computers MDPI

Article

Adaptive Contextual Risk-Based Model to Tackle Confidentiality-Based Attacks in Fog-IoT Paradigm

Satiaseelan Selvan and Manmeet Mahinderjit Singh *

School of Computer Sciences, Universiti Sains Malaysia, Gelugor 11800, Malaysia; seelan.043@live.com
* Correspondence: manmeet@usm.my

Abstract: The Internet of Things (IoT) allows billions of physical objects to be connected to gather and exchange information to offer numerous applications. It has unsupported features such as low latency, location awareness, and geographic distribution that are important for a few IoT applications. Fog computing is integrated into IoT to aid these features to increase computing, storage, and networking resources to the network edge. Unfortunately, it is faced with numerous security and privacy risks, raising severe concerns among users. Therefore, this research proposes a contextual risk-based access control model for Fog-IoT technology that considers real-time data information requests for IoT devices and gives dynamic feedback. The proposed model uses Fog-IoT environment features to estimate the security risk associated with each access request using device context, resource sensitivity, action severity, and risk history as inputs for the fuzzy risk model to compute the risk factor. Then, the proposed model uses a security agent in a fog node to provide adaptive features in which the device's behaviour is monitored to detect any abnormal actions from authorised devices. The proposed model is then evaluated against the existing model to benchmark the results. The fuzzy-based risk assessment model with enhanced MQTT authentication protocol and adaptive security agent showed an accurate risk score for seven random scenarios tested compared to the simple risk score calculations.

Keywords: fuzzy logic; risk assessment model; confidentiality-based attacks; security; sensors; Internet of Things; fog computing

check for updates

Citation: Selvan, S.; Mahinderjit Singh, M. Adaptive Contextual Risk-Based Model to Tackle Confidentiality-Based Attacks in Fog-IoT Paradigm. *Computers* **2022**, *11*, 16. https://doi.org/10.3390/computers11020016

Academic Editors: Paolo Bellavista, Kiran Kumar Pattanaik and Sourabh Bharti

Received: 25 August 2021
Accepted: 19 November 2021
Published: 24 January 2022

1. Introduction

A growing number of physical objects are being connected at an unprecedented rate, realising the idea of the Internet of Things (IoT) [1]. It is the internetworking of various objects and network connectivity that allows these objects to communicate and exchange data, including sensors, smart meters, smartphones, smart vehicles, radio-frequency identification (RFID) tags, personal digital assistants (PDAs), and other items such as embedded devices, software, and actuators. Some IoT applications might need a fast response, some might involve private data, which should be stored and processed locally, and some might produce large volumes of data, which could be a heavy burden for networks [2]. With the advance of IoT, fog computing [2,3], has been introduced to bring services closer to the end-users by pooling the available computing, storage, and networking resources at the edge of the network. However, the Fog-IoT network has more significant challenges because it has three different layers, and each layer requires a different technique of security assessments.

The target of this study is to tackle confidentiality-based attacks in the Fog-IoT network. Several problems are detected in the Fog-IoT network, which are man-in-the middle attacks, in which hackers control and manipulate the private communications of users, data leakage, and data manipulation and the existing access control models that are incapable to evolve according to the environmental changes.

In order to mitigate these problems, an enhanced MQTT protocol authentication method in the Fog-IoT network is being proposed. The authentication protocol covers a contextual risk model to continuously quantify risk on any fog node configured to join the network. In ensuring that the abnormality action on the devices is captured, an adaptive monitoring security agent in the fog node is also proposed. The research objectives mapped with the following research questions: (i) What type of method is needed to tackle confidentiality-based attacks in the Fog-IoT network? (ii) What model is suitable to quantify risk in fog nodes? (iii) What technique is needed to detect abnormal actions from the device?

Several contributions are made. First, an enhanced MQTT protocol authentication method in the Fog-IoT network is developed and tested. The researcher used the EMQX MQTT broker to enhance the authentication method since the broker is a scalable and highly extensible MQTT broker. The broker allows pre-trigger as a plugin for client authentication. The researcher used the plugin to configure the command to check the username, IP address, and risk value against the MYSQL database, which is shared with the access control mechanism. A unique token is introduced in the authentication, which will be secret between the sender and receiver in the communication channel.

The next contribution is presenting a contextual risk model to quantify risk in fog node using a fuzzy inference system. Two types of risk estimation methods are created to quantify the risk, risk assessment, and fuzzy model. Based on the existing models, the proposed model is enhanced to support adaptive features by adding the history and device context to better estimate risk. The risk assessment model is used with a Multifactor Evaluation Process to provide precise risk calculation while the fuzzy model approaches enhanced with a risk assessment formula using the MATLAB tool. Both models are simulated in the Fog-IoT setup and evaluated. The risk factor results are compared with a series of scenarios to get accurate results. The enhanced fuzzy model provided better results in seven different random scenarios testing with the same Fog-IoT setup using the risk assessment.

The last contribution is an adaptive monitoring security agent in the fog node capable of detecting abnormal actions from the device. The adaptive monitoring security agent developed in the fog node and an actual fog node performs filtering from clients. The agent constantly monitors for unauthorized access or connection-related log information from the EMQX MQTT broker. The agent updates client info as a blacklisted device in the MYSQL database if a particular client cannot authenticate. The agent also terminates any connected client in the channel by rerunning the access control of the EMQX MQTT broker. The adaptive monitoring security agent is simulated in the normal authentication method with valid and invalid client info. The agent also tested the security analysis of the Man-in-the-Middle attack scenarios.

In evaluation model, several case studies demonstrating calculated risks from the model are presented. One notifiable contribution to the body of knowledge is proving the essence of adaptive risk and monitoring mechanism in the Fog-IoT environment. Due to the instability and insecurity that lies in IoT sensors nodes and its communication protocols, the proposed idea clearly demonstrated the usage of the MQTT protocol and risk model in initiating any new connection.

The remainder of the paper is organised as follows: Section 2 discusses the literature review. Section 3 presents in-depth information on the methodology used in developing the proposed work. Section 4 presents the results. Section 5 presents the discussion related to the proposed solution and results, and Section 6 presents the conclusion.

2. Background and Literature Review

Fog computing is the worldview broadening of cloud computing and its administrations to the edge of the system. Fog is recognised from the cloud in its nearness to end-clients/hubs, thick land circulation, support mobility, heterogeneity, interoperability, and pre-preparing. Fog computing does not replace cloud computing, but it supplements cloud computing and intends to give a computing and capacity stage physically nearer to the end hubs, provisioning other types of uses and administrations with a productive exchange with the cloud layer [3]. Fog computing is a virtualized platform that offers computational, networking, and storage services between cloud computing and end devices. The features of fog computing are as follows:

- Decentralisation—fog nodes are an independent entity which can manage the resource and services and do not require a centralised server to control the system [4].
- Low latency—fog node is closer to the Fog-IoT end devices. Thus the latency is lowered compared to when accessing to the cloud directly [4].
- Large scale IoT applications support—Fog computing supports large-scale IoT applications such as environmental monitoring and climate change monitoring, which bring heavy management overhead to the centralised cloud.
- Heterogeneity—fog computing is a virtualised platform that offers computational, networking, and storage services between cloud computing and end devices. It serves as a building block, as it exists in different forms, and can be deployed in wide-ranging environments.
- Interplay with cloud—cloud computing is a central global entity, and the fog node focuses on localised data processing. Big Data analysis and machine learning use both cloud and fog computing to perform data analysis.
- Predominance of wireless access—cloud layer provides global coverage, which is used as a repository to store years of data. This information is typically used in machine learning analysis and metric systems using indicators.
- Online analytics—many applications require both Fog localisation and Cloud globalisation, mainly for analytics and Big Data. This first tier of the Fog, designed for machine-to-machine (M2M) interaction, collects and processes the data and issues control commands to the actuators. It also filters the data to be consumed locally and sends the rest to the higher tiers. The second and third-tier deal with visualisation and reporting (human-to-machine (HMI) interactions), as well as systems and processes (M2M). The timescales of these interactions range from seconds to minutes (real-time analytics) and even days (transactional analytics). Some data relates to protection and control loops that require real-time processing (from milliseconds to sub-seconds).
- Mobility support—fog computing applications enable direct communication with mobile devices using protocols such as Cisco's Locator/ID Separation Protocol that decouples host identity from location identity using a distributed directory system.
- Geographic distribution—fog computing must support different storage levels from the lowest tier to the highest tier. The higher the level, the broader geographical coverage, and the longer the time scale.
- Location awareness—the location of fog nodes can be traced actively or passively to support devices with rich services at the network edge. Fog computing dedicates to local IoT applications accessible for the devices in certain areas via specific fog nodes. Therefore, it is aware of the devices' regions based on the locations of fog nodes [4].

Since technology evolves rapidly, most systems do not have the intelligence to detect suspicious behaviour and mostly block malicious activity. The software protection solution needs to change from restricting malicious activity to actively monitoring the behaviour of the system. There are many ways to lower the attack's impact, such as shrinking the attack surface to prevent threats from spreading, segmenting the system at the network level, and reducing the remediation time by acting promptly to all the attacks and slowing down the rate of the attack. This can be achieved with adaptive security architecture (ASA). Figure 1 shows the architecture of adaptive security.

Based on Figure 1, the four aspects of ASA are prediction, prevention, detection, and response. The leading role of the prediction stage is to analyse the visibility of threats, predict the threats, and expect any new security attacks to the current system. The future attack scenarios are predicted from intelligence from history and other environment states. Next, in the prevention stage, the potentials threats are prohibited. The system is isolated to protect against any threats by reducing the surface area of potential attacks. The information on malicious activities is captured in the system. Then, in the detection stage, it continuously monitors the system, which enables the detection of attacks as early as possible. The risk factor associated with the target is assessed and prioritised based on the detection incident. On the other hand, the system can isolate the affected part due to the potential attack to prevent any further critical damage. Finally, it is the response stage which responds to an attack. The attack needs to be analysed to get in-detail information about the full scope of the breach. Thus, adaptive security architecture can be implemented in mitigating security attacks.

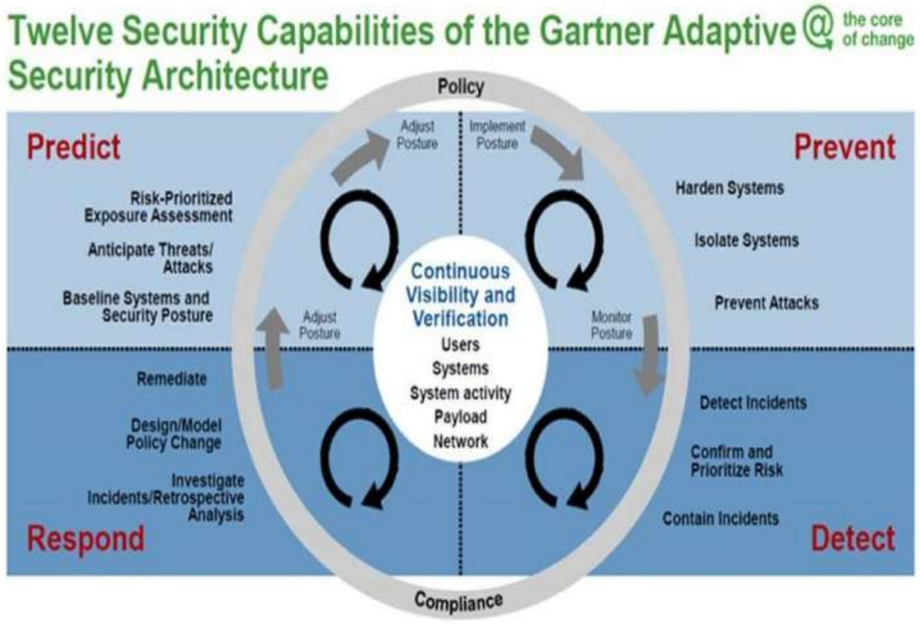

Twelve Security Capabilities of the Gartner Adaptive Security Architecture
@ the core of change
Policy
Compliance
Predict
Prevent
Respond
Detect
Risk-Prioritized Exposure Assessment
Anticipate Threats/ Attacks
Baseline Systems and Security Posture
Harden Systems
Isolate Systems
Prevent Attacks
Remediate
Design/Model Policy Change
Investigate Incidents/Retrospective Analysis
Detect Incidents
Confirm and Prioritize Risk
Contain Incidents
Adjust Posture
Implement Posture
Adjust Posture
Monitor Posture
Continuous Visibility and Verification
Users
Systems
System activity
Payload
Network

Figure 1. Adaptive security architecture [5].

2.1. Security Attacks

In this section, comprehensive security issues in the Fog-IoT network are discussed. The security attacks are classified into three levels: node-level attacks, network-level attacks, and application-level attacks [6]. These attacks can be prevented and mitigated by applying several countermeasures. Table 1 shows the level of attacks, its type of attacks, and the countermeasures that can be applied.

Table 1. Classification of security attacks in fog computing [6].

Level of Attacks	Types of Attacks	Countermeasures
Node-level attacks	Node capture attacks Node jamming Malicious code injection on node Physical damage of the node RF interferences and Eavesdropping False data injection attacks Replay attacks Crypt analysis attacks Sleep deprivation attacks	Device Authentication Data Confidentiality Data Integrity Secure booting by using low power cryptographic hash functions Data Anonymity Access Control Devices Physical barriers Monitoring devices
Network-level attacks	Denial of Services (DoS) Spoofing attacks Sinkhole attacks Man-in-the-Middle attacks Routing information attacks Sybil attacks Unauthorised access RFID cloning Traffic analysis attacks	Network Authentication mechanisms Confidentiality and integrity of transmitted data Implementation of routing security Secure user data on devices by using encryption and cryptographic mechanisms

Table 1. *Cont.*

Level of Attacks	Types of Attacks	Countermeasures
Application-level attacks	Phishing attacks Malicious Virus/Worms Trojan horse Ransomware Spyware	Access control lists Firewalls Protective softwares Intrusion detection mechanisms Trust management

These security attacks can be prevented through the implementation of an access control mechanism. The following section discusses the access control mechanism.

2.2. Access Control in Fog-IoT

The Fog-IoT devices exchange multiple information related to the state of the environment or the end users' behaviour. The communication channel between the Fog-IoT system is crucial, and a robust access control mechanism is needed to prevent any security attacks. Figure 2 shows the hierarchical process of access control according to each Fog-IoT layer.

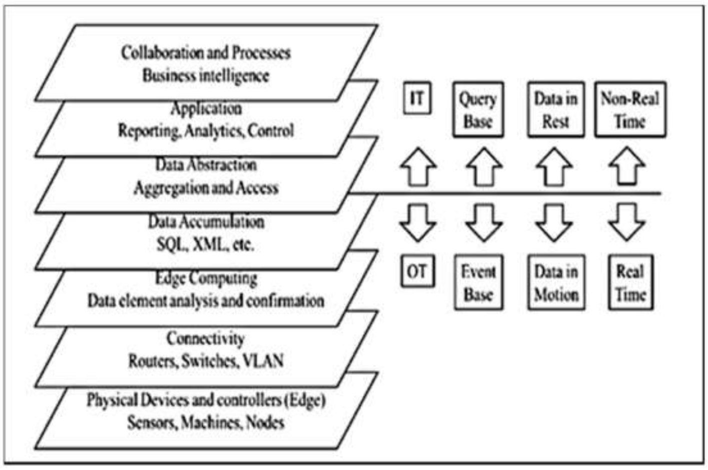


Figure 2. IoT flow model.

The main motive of access control is to allow access to allowed users and prevent authorised users from accessing the system in an unauthorised manner [7]. This is done to ensure that only authorised users are allowed to access a system to protect confidentiality and integrity. There are two types of access controls which are traditional and dynamic access control [8].

Traditional access control is rigid and uses static access policies. The outcome of the policies is always the same regardless of any situation because they are predefined. When there are changes in the state of the environment, the static approach fails to adapt to the condition for access decision-making [4]. For instance, Discretionary Access Control (DAC), Mandatory Access Control (MAC), and Role-based Access Control (RBAC) are the three main traditional access control models.

Dynamic access control is a very flexible model that uses dynamic contextual features together with access policies. The decision-making happens in real-time at the time of the request [9,10]. For instance, real-time features can be trust, risk, context, history, and operation need [11,12]. For this study, we proposed a contextual risk-based access control model that uses the security risk factor as one of the inputs for access decision making.

2.3. Risk-Based Access Control Model

The risk factor can be estimated for each request in a system that allows or denies access dynamically [4]. The access decision is made by performing a risk analysis on each user access request [13]. The formula to represent quantitative risk is Quantified

Risk = Likelihood × Impact (1). The probability of an incident is the likelihood, and the impact is the estimation of the damage over any incident [14]. There are two types of quantified risk-based access control: non-adaptive and adaptive. The adaptive model requires a continuous monitoring process, and the risk estimation is adaptive based on access sessions. In contrast, the non-adaptive model measures the risk during each request and does not continuously monitor abnormal detection [15].

Many improvements can be made in the access control in the Fog-IoT network. Several existing risk-related models that use the security risk for dynamic access control models are discussed. In order to fulfil the gaps found in existing models, this study implements adaptive security architecture in access control decision-making to strengthen the security solution. Table 2 shows the comparison between existing risk-based access control models.

Table 2. Comparison between existing risk-based access control models.

Author(s)	Risk Estimation Technique	Risk Factors	Limitations
[16]	Risk Assessment	Outcomes of actions	Risk factors are limited and lack of adaptive approach
[17]	Fuzzy MLS Model	Subject security level and object security level differences	History of user behaviour not used to predict the future
[18]	Fuzzy Model	Data sensitivity, action severity, and user risk history	No clear risk estimation and lack of adaptive approach
[19]	Fuzzy Interference	Object security level and subject security level	Inefficient fuzz inference and lack of adaptive method
[20]	Risk Assessment	Object sensitivity, subject trust, and the difference between them	Past risk history not counted in the access control model
[21]	Risk Assessment	History of reward and penalty points	No predictive risk technique used
[22]	Game Theory	Access benefits of the subject	Risk factors are limited, and the adaptive approach not used
[23]	Mathematics Functions	Data Sensitivity, action severity, and risk history	Lack of risk prediction method
[24]	Risk Assessment	Outcomes of actions	Risk factors are limited and lack of risk prediction technique
[25]	Mathematics Functions	Risk policies	Risk factors are limited and lack of prediction

Based on the risk model analysis performed by previous researchers, several limitations have been captured, such as no history factor [16–18] being accounted for in risk calculations, no prediction in risk calculations [19–21], and no contextual factors presenting the current state of environment being included in risk calculations [16,18,22–25]. Thus, the need to propose a new contextual risk model in essential. One interesting study showing how the proposed cloud-edge ecosystem model is being tested is shared by Ficco et al. (2017) [26]. However, instead of testing in an emulated environment, we would only test our proposed model in a simulated environment. The reason for this being that, since this work is at the experimental stage, a simulation may be sufficient. Nevertheless, the option to emulate the model would be achieved.

2.4. Authentication in Fog-IoT

In this section, the authentication process is discussed in detail. This study uses the Message Telemetry Transport (MQTT) protocol to set up the Fog-IoT network. The protocol is ideal for machine-to-machine interaction as it uses fewer system resources such as low power and less data packet size. The MQTT protocol provides a basic authentication mechanism. Due to this reason, the EMQX MQTT broker is considered to enhance the authentication mechanism. The types of authentications are discussed in Table 3.

Table 3. Techniques of authentications.

Techniques of Authentication	Descriptions
Username and password authentication	▪ Similar to basic MQTT protocol authentication technique ▪ Pre-trigger uses a configuration file that has username and password information ▪ Allows defining multiple clients for a system.
MySQL authentication	▪ MySQL database is used as a medium in the authentication step. ▪ EMQX generates the SQL command with the input parameter to check against the stored data. ▪ MySQL database is available before the authentication process. ▪ For instance, the value of the password field of a query result is compared with the input client password after salt encryption, whether it is a match or vice versa. The client can be authenticated if the password field exists as a result.
Redis authentication	▪ When a client is up and running, the Redis authentication pre-trigger is connected to the Redis server. ▪ The authentication info is validated with the stored data, and this determines the client access connection. ▪ The client data are stored before the Redis service starts.
MongoDB authentication	▪ One of the techniques used as a pre-trigger. ▪ The information stored in the MongoDB database are re-used during the authentication phase. ▪ This determines access to a particular client.

Evaluation of MQTT Authentication

The EMQX broker provides various technologies to enhance the authentication scheme. MySQL authentication is apt for the current research because there are many use cases for a relational database such as MySQL. Any applications that require multi-row transactions are better suited for a relational database. MongoDB has a greater challenge to replace legacy systems that were built for relational databases. Arduino and Node.js also use MySQL databases, and the libraries are easily implemented. In this study, the MySQL plugin is combined with the MySQL technique to support risk factors and introduce strict token rules to authenticate the devices, making the system robust. The methods are further discussed in Section 3.

3. Research Methodology

In this section, the research methodology of the proposed approach is explained phase by phase, as each phase has contributed an output which indirectly is the pre-request for the next phase in the research cycle. Figure 3 shows the phases involved in this research.

Based on Figure 3, there are four phases which are: analysis (phase 1), design (phase 2), simulate (phase 3), and evaluate (phase 4). Starting with phase 1, a literature review is conducted on the previous work regarding Fog-IoT and its security challenges. In analysing the requirements, some research on the existing risk models based on context awareness criteria such as location, time, network type, and the user is conducted. Contextual awareness is the ability of a given application to access information about the physical environment and automatically adapt its behaviour appropriately in real-time. Based on the criteria stated, each parameter reveals abnormal or unintentional conditions. The system can detect the changes in the environment state with the help of device context and reacts actively in real-time. The locations reveal the latitude and longitude of the devices that are connected to the system. Time is also an essential criterion for a sensor device to be activated in a system that reveals the usage timeline. The information of the selected network type, where a device communicates through the gateway, is captured because it helps in filtering intruders in a system. Besides, the device ID helps to determine whether it belongs to the allowed list or vice versa. Based on the study conducted, the major problem in Fog-IoT is the compromise in data confidentiality [27,28]. Diverse forms of security attacks are identified in fog network layers. This study analyses several dynamic risk models based on security attacks and a literature review on the MQTT authentication methods.

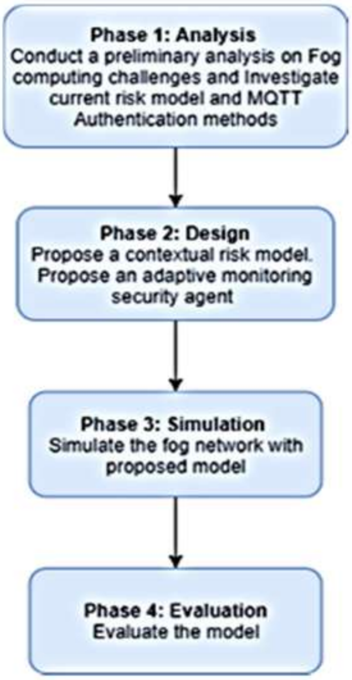

Phase 1: Analysis
Conduct a preliminary analysis on Fog computing challenges and Investigate current risk model and MQTT Authentication methods
Phase 2: Design
Propose a contextual risk model. Propose an adaptive monitoring security agent
Phase 3: Simulation
Simulate the fog network with proposed model
Phase 4: Evaluation
Evaluate the model

Figure 3. Research phases.

In phase 2, which is design, the technique is planned from the data collected from phase 1, and the problem is identified. This stage identifies the project requirement and the techniques of enhancement. A contextual risk-based model is proposed based on the analysis of access control models and adaptive monitoring security agents in the fog node to mitigate the data confidentiality-based attacks [29]. The proposed adaptive system based on adaptive security architecture consists of three main elements. First is the ability of the designed system to analyse and predict the vulnerabilities, the potential threats, and anticipate new types of attacks in the existing system. The information from history and other sources are used to anticipate future attack scenarios. The access control mechanism stores the access list of devices associated with risk in the database to predict behaviour for the future. Second, there is a method of prevention. The system is hardened and isolated to protect against the identified threats by reducing the surface area for potential attacks. Known malicious activities are countered by using the blacklisting approach based on past incidents. The access control terminates the connection of a device based on the prediction step. Third, the component of an adaptive method comes into the picture. Continuous system monitoring enables the detection of a successful attack as early as possible. After the detection of an incident, the risk associated with it is assessed and prioritised. The security agent in the fog node monitors the behaviour of the successful communication of a device. If a connected session takes a longer time than expected, it results in connection termination. The device info and its risk value are adjusted accordingly. The risk estimation value is calculated based on previous researchers' formula [20].

In phase 3, which is a simulation, a secure and non-secure Fog-IoT network to simulate the proposed adaptive contextual risk model are set up. Figure 4 shows the Fog-IoT secure setup with four layers. First, the end device layer consists of sensors and an IoT platform. The NodeMCU with temperature and humidity sensors are used. Second, the gateways and switching layer push the data from end devices to the fog nodes or receive the data from the fog node to the end devices. Third, the fog layer has three segments: Fog Node Manager, known as an MQTT broker, adaptive monitoring agent, and actual fog node, which analyses based on the data. The proposed model is part of the fog node. The last layer is the cloud, which stores the data permanently.

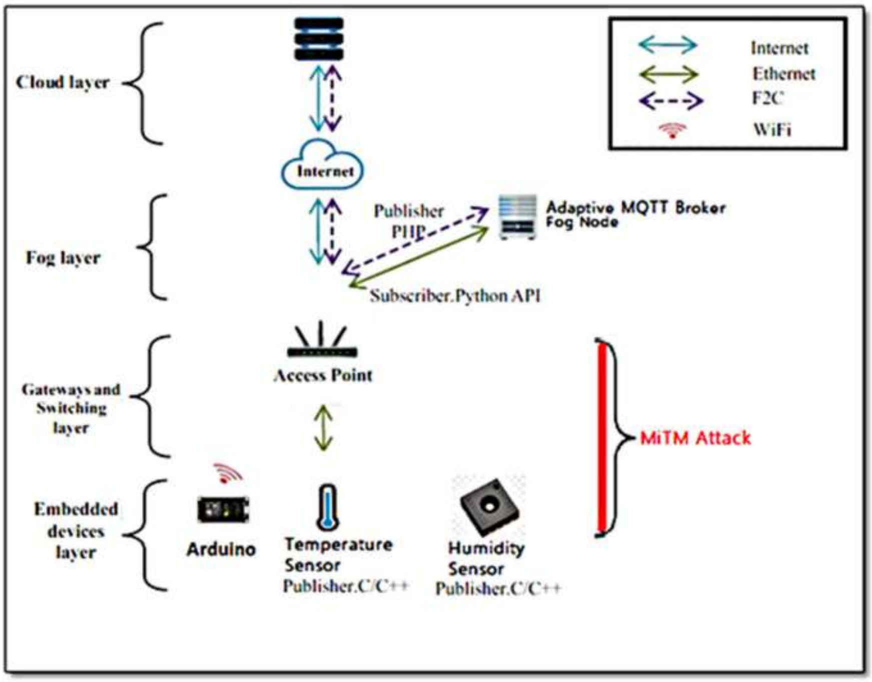


Figure 4. Fog-IoT simulation setup.

Another setup is similar to the earlier version but with the inclusion of a Man-in-the-Middle attack between the fog layer and the end devices. Figure 5 shows the modified simulation setup to measure the proposed algorithm and improve the technique if needed. The results are captured and evaluated in the last phase. The final phase is evaluation. For the evaluation purpose, each mentioned objective is evaluated based on the evaluation metrics. The evaluation metrics are further discussed in the following subsection.

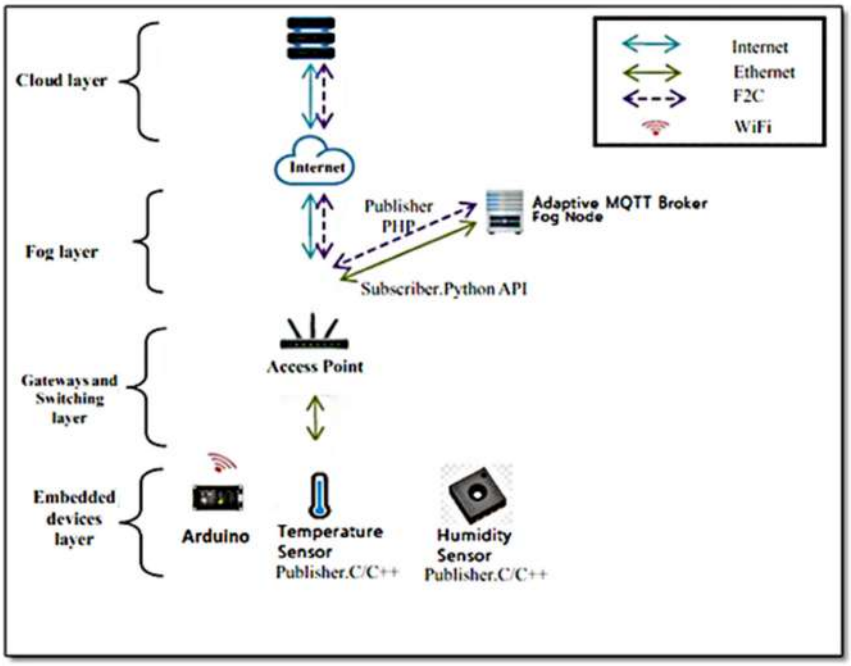


Figure 5. Modified Fog-IoT simulation setup.

3.1. Evaluation Metrics

According to the evaluation done in the final phase, the evaluation is done based on the evaluation metrics. The in-depth discussion is presented in this section.

3.1.1. MQTT Plugin vs. MQTT

A literature review on the MQTT protocol and its authentication type is conducted. The MQTT protocol provides basic authentication using a username and password scheme and the module is rigid to adapt the proposed risk model. EMQX broker provides additional control for authentication using the MySQL database. The integration of the emqx_auth_mysql plugin and MQTT offers additional control to the system. Thus, the EMQX MQTT plugin is used.

3.1.2. Risk Model and Factor Quantification

The proposed contextual risk-based model is simulated in Fog-IoT networks and evaluated using the risk assessment formula. Figure 6 shows the risk scale used to compare with the risk factor obtained through simulation.

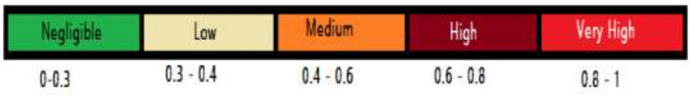


Figure 6. Risk scale.

Based on Figure 6, the output of the risk factor is compared against the risk quadrant with a threshold value of 0.3. The threshold value is set based on risk assessment by [20]. The result of every communication step for the same device varies according to the simulating condition. The presence and absence of an adaptive security agent also influenced the risk factor of the access control.

3.1.3. Case Scenario of Testing the Proposed Model against Man-in-the-Middle Attack

Testing and implementing the Man-in-the-Middle attack in the normal condition was conducted to differentiate the outcomes. The security analysis is performed by changing the input of the proposed model and the conditions. The resource allocation and network performance are also evaluated using the proposed model. The evaluation also contributed to some opinions on the future enhancement of this study.

4. Adaptive Contextual Risk-Based Model

The proposed contextual risk model illustrates the complete interaction of the Fog-IoT model, which covers MQTT authentication between devices and server, risk-based access control, and adaptive monitoring agent to monitor the network. The proposed model has been set up in an environment shown in Figure 5. The proposed adaptive MQTT broker fog node aims to calculate the risk of connections based on authorised credentials and several contextual rules triggered by the fuzzy engine. The following section explains the operation flow of the proposed model.

4.1. Proposed Contextual Risk-Based Model Operation Flow

In this study, the Fog-IoT architecture is designed to implement the proposed adaptive security architecture. Figure 7 shows a simple IoT environment operation flow setup with connections between cloud, fog computing, and end devices or sensors. The proposed fog layer has two segments which are the adaptive MQTT broker and the actual fog node. The sensors' data are published to the EMQX broker using the wireless access point. The NodeMCU microcontroller aggregates the temperate and humidity sensor and publishes two times per minute. The data is subscribed by the fog server using the same wireless access point. The data filtering happens in the fog server, and only the distinct results are stored in the cloud server. An estimated total of 10 packets are sent per 5 h.

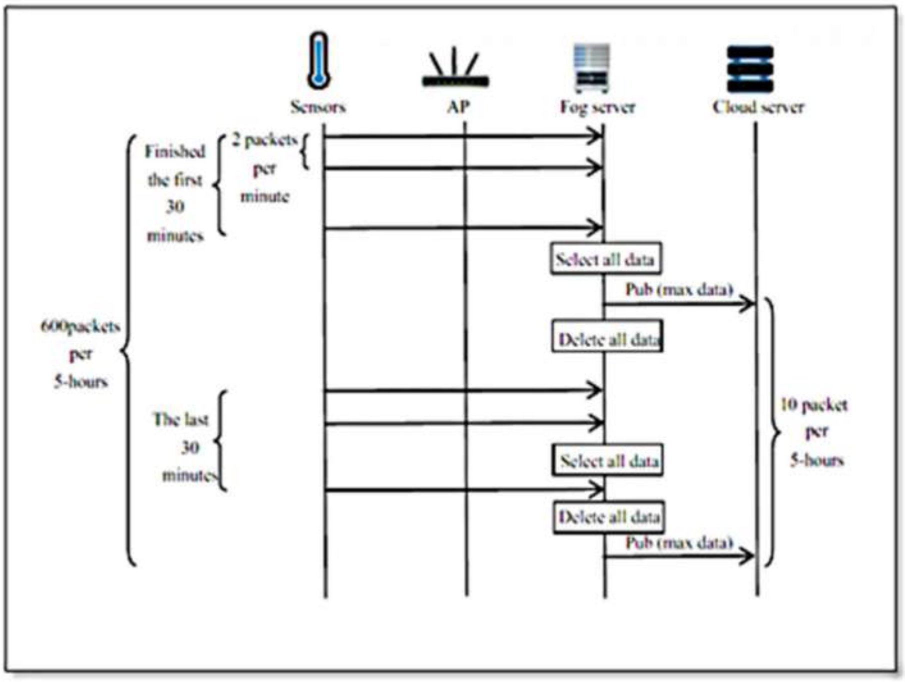


Figure 7. Proposed Method Operation Flow.

Based on Figure 7, there are four layers: end devices, gateway and switching, fog node, and cloud. Figure 8 shows the components involved in the adaptive MQTT. Each layer and the respective components are explained in detail as follows:

- End Device Layer—this layer is similar to the physical layer of the Open Standard Interconnection (OSI) model. It consists of sensors, actuators, and microcontrollers. The motive of this layer is to gather data from IoT devices, and then these data are transmitted to fog nodes through a communication channel such as WiFi. These devices are limited with a resource such as power and storage. The data or the connection can be monitored and managed in the same Local Area Network (LAN) using a monitoring dashboard or tools. In this study, the temperature and humidity sensor is used to test the Fog-IoT. The sensors are plugged into the NodeMCU IoT platform, and the data are transmitted using a WiFi network. MQTT protocol version 5.1.1 is used in this layer to publish the data. The publisher is programmed using the C/C++ programming language by Arduino IDE.
- Gateway and Switching Layer—the motive of this layer is to transmit the information from the end devices to the fog node or obtain the response from the fog node to the end device. In this study, the NodeMCU with a dht22 sensor is connected to the fog layer using the IEEE802.11n access point.
- Fog Node Layer—this layer consists of three sections: Fog Node Manager, known as an MQTT broker, the adaptive monitoring security agent, and the actual fog node, which are analysed based on the data. This layer considers the "middle man" and obtains the temperature and humidity data from the dht22 sensor. Then, the fog server does the data filtering and sends the data to the cloud server. The fog node uses the EMQX broker to subscribe to the data from the sensors using the Node.js Application Program Interface (API). These data are temporarily stored in the fog server before sending it to the cloud. EMQX broker applies the adaptive security framework to authenticate the clients.
- Cloud Layer—this layer aims to store important data permanently. In this study, the Google cloud server database is used to store the data. The data are pushed from the fog layer whenever needed.

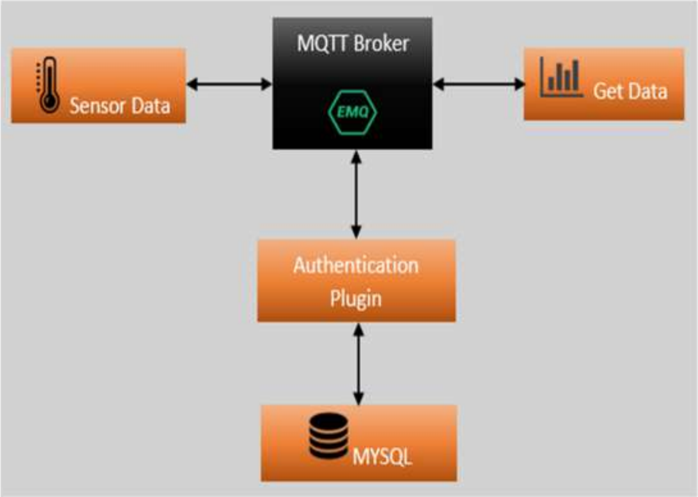


Figure 8. Components of Adaptive MQTT.

4.2. Adaptive MQTT Broker for Communications

MQTT is a lightweight protocol often used for devices to communicate with other systems. It is used to publish and subscribe messaging techniques to exchange data. Figure 8 shows the components involved in adaptive MQTT broker.

Based on Figure 8, this system has five components: a publisher of messages (sensor data), a subscriber to messages (client), a broker that connects the sensors and the clients, an authentication plugin, and a MySQL database. Each component is explained in detail as follows:

- Broker—in this study, the EMQX broker is used to set up the localhost machine. This is to allow for the complete control of the broker. The broker provides a web dashboard and backend access. NodeMCU with dht22 sensors are connected directly to the broker and can publish the temperature and humidity data. The fog server can authenticate the broker to obtain the data.
- Authentication Plugin—the MQTT server controls the client access authentication step. The EMQX allows authentication setup in a different layer, such as the transport layer and application layer. It allows the support of different layers. The MySQL authentication technique is used to enhance the current authentication scheme. MySQL script plugin is used to set up the MySQL database connection and query filtering. The SQL query provides results, and the results are validated against the rules check in the configuration file. The broker uses a hardcoded token to check the allowed device list to authenticate. This token is used by the device as well; if the selected user exists and the risk is less than 0.3, the device can authenticate.
- MySQL—the devices and the broker communicate with this user table, which contains the username, password, risk factor value, and IP address to validate the info.
- Publisher—after the MQTT client establishes the connection with the broker, the client can publish messages. MQTT filters the messages using a topic on the broker. Every message contains a topic which the broker helps to send to the subscribed clients. Every MQTT packet payload contains the data to transmit in byte format. The use case of the client decides on the payload structure. The client can publish the data in binary data, text data, or even XML or JSON. NodeMCU uses the dht22 sensor to establish a connection and publish the data EMQX. The MQTT packet of a published message has several attributes.
- Subscriber—the client receives the data from the broker known as a subscriber. A subscriber usually presents in an IoT network as some entity that needs to acquire the data. The client needs to send a subscription message on the topic of interest to the MQTT broker. The subscription message contains a unique identifier and a list of subscriptions— in this research, the client subscription to the fog server. Once the

data is obtained, it performs the data filtering algorithm and selectively pushes it to the cloud.

4.3. Access Control Construction with Risk Assessment

Risk assessment can be an additional factor in access decision-making. In security analysis, it can be challenging to pick the correct path for every condition. With the additional risk criteria, good decisions are made. Based on [20], the risk assessment formula, the risk value is evaluated after calculating the cost of outcomes based on the relevant action and device context. The initial formula is enhanced to support additional inputs. These are three main risk values of the outcomes: availability, integrity, and confidentiality. The formula for each risk value of the outcome is shown as follows:

$$\mathrm{RV_A}(\mathrm{O_{ai,j}}) = \mathrm{C_A}(\mathrm{O_{ai,j}}) \times \sum\nolimits_k \int \mathrm{O_{ai,j}}, \mathrm{S_k} \tag{1}$$

$$\mathrm{RV_I}(\mathrm{O_{ai,j}}) = \mathrm{C_I}(\mathrm{O_{ai,j}}) \times \sum\nolimits_k \int \mathrm{O_{ai,j}}, \mathrm{S_k} \tag{2}$$

$$\mathrm{RV_C}(\mathrm{O_{ai,j}}) = \mathrm{C_c}(\mathrm{O_{ai,j}}) \times \sum\nolimits_k \int \mathrm{O_{ai,j}}, \mathrm{S_k} \tag{3}$$

The notion used in equations are as follows:

- A sensor or a device that communicates with other devices denotes a client by S.
- Client, S, executes an action, A.
- An outcome, O, of an action, A, is a consequence.
- Context, k, is information related to the current action.
- Consequence, c(O), is a function for calculating the cost of each outcome, O.
- Risk, RV (O, a), is a function for calculating risk value.

Based on the three equations shown, Equations (1)–(3) are used to calculate the risk value of the outcome in terms of availability, integrity, and confidentiality, accordingly. From these calculations, the risk factor can be calculated using the following final formula, where $Wi \in \mathrm{N}$, I = 1, 2, 3, 4, 5, 6, 7, 8. It can be adjusted according to suitable values based on the weight of a specific metric.

$$\mathrm{RV} = \frac{\mathrm{W_1\ RV_A + W_2\ RV_I + W_3\ RV_C + W_4\ RV_T + W_5\ RV_N + W_6\ RV_D + W_7\ RV_L + W_8\ RV_{PR}}}{\mathrm{W_i}} \tag{4}$$

Based on Equation (4), the final risk factor can be calculated by mapping the weighted arithmetic mean with the risk factor of availability, integrity, and confidentiality. There are a few steps involved in risk assessment. First, the client's action–outcome is captured. The action can be viewed, created, modified, or deleted. Next, the action is mapped with each factor's availability, integrity, and confidentiality. The weight is assigned for each factor. Then, the cost of the impact of availability, integrity, and confidentiality are determined. Next, based on the set of device context and action, the probability is calculated. Finally, the threshold risk value is obtained, which is used for comparison purposes. If the risk factor is less than 0.3, the device is allowed to authenticate. MFEP is used with the input parameters to obtain a better solution.

4.4. Access Control Construction Using Fuzzy Logic

There is a lot of uncertainty present during the risk assessment, and it affects the calculation of risk in a scalar; for instance, when identifying the probability of impact with severity and mathematics. Fuzzy logic calculates the results using the degree of truth. It can represent nonlinear functions of arbitrary complexity. Based on researcher [30], the fuzzy approach is enhanced with the risk assessment formula to compute the risk factor. Figure 9 shows the fuzzy risk model.

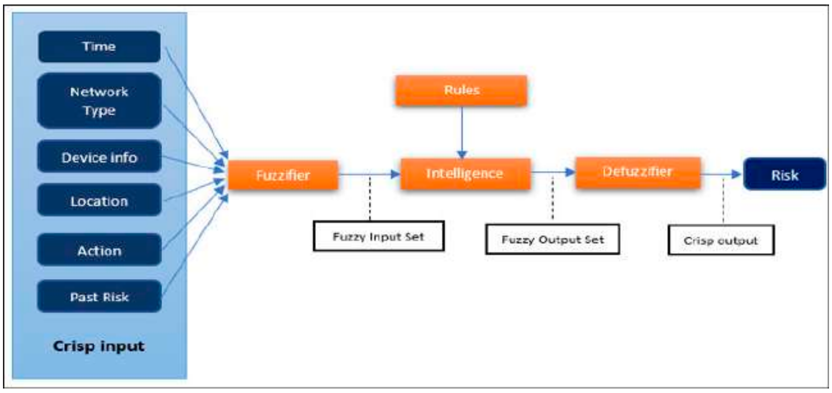


Figure 9. Fuzzy risk model.

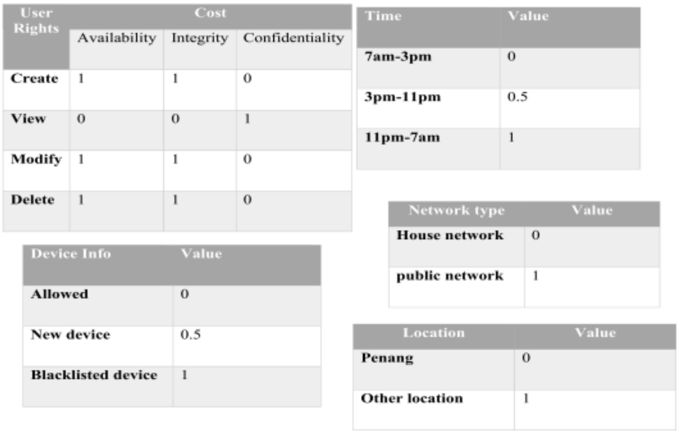
Based on Figure 9, there are four main steps involved in the fuzzy risk assessment. The first step is the identification of the key risk indicators that define the linguistic variables. The input and output variables, and their ranges, are also identified. There are six indicators specified as input: time, network type, device info, location, action, and past risk. Secondly, the ranges for the input are set accordingly and separated into classes or fuzzy sets. Thirdly, the fuzzy rules are defined in this step. The risk varies on the factors and needs to specify how the risk varies as a function of the factors. For instance, action severity can range from low to high. Finally, the fuzzy model is encoded, and the system is tuned using the MATLAB tool.

5. Results and Findings

In this section, the results and findings are presented. The risk model evaluation and security analysis findings are discussed.

5.1. Evaluation of Proposed Contextual Risk Model

Two models are discovered based on the analysis, risk assessment, and fuzzy-based approach. The risk assessment method provides a precise way to compute the risk, whereas the fuzzy approach is applied using the risk assessment to quantify the risk factor. Fuzzy modelling of risk factors addresses uncertainties in real applications, which can improve the performance of the risk assessment methodology. Based on the formulas expressed in Section 4.3, and using the defined input values of time, network type, device info, and location, using normalised range as shown in Figure 10, one example shows when a user needs to add new sensor data within a Fog-IoT environment. The input variable for the fuzzy table is presented in Table A1 in Appendix A.



User Rights	Cost		
	Availability	Integrity	Confidentiality
Create	1	1	0
View	0	0	1
Modify	1	1	0
Delete	1	1	0

Time	Value
7am-3pm	0
3pm-11pm	0.5
11pm-7am	1

Network type	Value
House network	0
public network	1

Device Info	Value
Allowed	0
New device	0.5
Blacklisted device	1

Location	Value
Penang	0
Other location	1

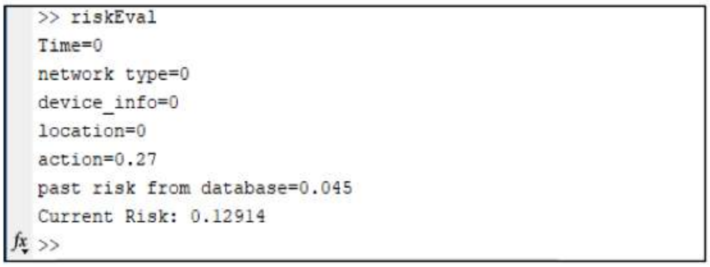
Figure 10. Data Input for Fuzzy Set.

The input values of time, network type, device info, and location are set based on the normalized range as shown in Table A1. For instance, the input value of the action parameter uses the normalized AIC score to measure the severity of the impact, which is defined in the risk assessment model. Next, both input values and fuzzy variables are calculated to show the conceptual risk, which is shown in Table 4.

Table 4. Risk Values Generated.

Input Type	Risk Values
User Action	RV_A = 1 × 0.14
	RV_C = 1 × 0.13
	RV_I = 0 × 0.13
	RV_{ACI} = 0.27
Time	RV_T = 0 × 0.11
Home Network	RV_N = 0 × 0.11
Allowed/Blocked List	RV_D = 0 × 0.11
Location	RV_L = 0 × 0.11
Device Provenance	Device Past Risk = 0.3
	RV_{Pv} 0.3 × 0.15 = 0.045

By using fuzzy, the process is presented as follows: First, the degree of the function is determined based on the crisp input of RV_{AIC}, $RV_{T,}$ RV_N, RV_D, RV_L, RV_{PR}. The second step is taking the fuzzified input and applying the following rule: if RV_{AIC} is medium, RV_T is low, RV_N is low, RV_D is low, RV_L is low, and RV_{PR} is low, then the final risk is low. The unification of the output of all rules is carried out, which is called aggregation. The membership function of all rules is clipped or scaled and merged into a single fuzzy set. Figure 11 displays the overall evaluation score, which is 0.129.



```
>> riskEval
Time=0
network type=0
device_info=0
location=0
action=0.27
past risk from database=0.045
Current Risk: 0.12914
fx >>
```

Figure 11. Risk Evaluation Screen.

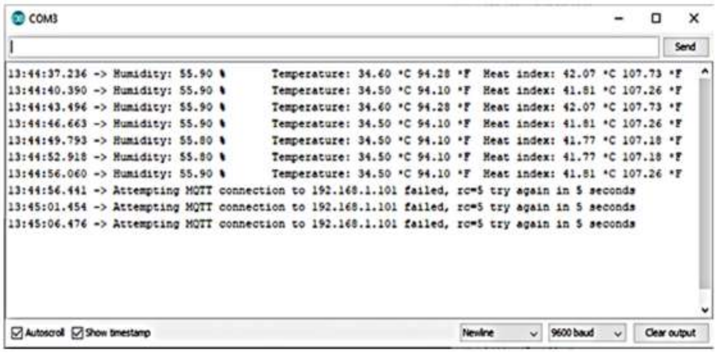
Next, the authors had simulated seven (7) types of random scenarios (as shown in Table 5). Case 1 and 4 risk factors are almost close, and other cases show that the delta of the two risk factors are much higher. In the current experiment, the fuzzy system shows a more reliable score. Thus, the fuzzy approach provides a more accurate score in contrast to simple risk score calculations.

5.2. Study of Security Analysis

An attacker can perform packet sniffing in the network by accessing the router/machine. Packet sniffing is the act of capturing packets of data flowing across a computer network. In most cases, an attacker can dissect the MQTT packet and can read the headers for the connection information. The MQTT packet information is not encrypted entirely. If a device uses a plain username and password, the attacker can obtain the value and other important information such as the broker info source and destination IP address. An attacker is able to read, insert, and modify the data in the intercepted communication. An attacker who tries to publish invalid temperature data by connecting to the MQTT broker is rejected. The broker rejected the connection because the ClientID of the connection and token does not tally with the username. This prevents an attacker from even authenticating to the broker behind an active connection. The EMQX server log captures this incident. Figure 12 shows the sensor connection when terminated once the new risk score is updated after the attack happens.

Table 5. Risk model evaluation result.

Case	Time	Network Type	Device Info	Location	Action	Past Risk	Risk Assessment Score	Fuzzy System Score
1	7 a.m.–3 p.m.	Public network	New device	Other location	Generate sensor data	0.34	0.35	0.41
2	7 a.m.–3 p.m.	House network	Allowed list	Penang	Generate sensor data	0.28	0.06	0.14
3	7 a.m.–3 p.m.	House network	Blacklisted device	Penang	Generate sensor data	0.67	0.17	0.36
4	3 p.m.–11 p.m.	Public network	Allowed list	Penang	Generate sensor data	0.34	0.19	0.15
5	3 p.m.–11 p.m.	House network	Blacklisted device	Penang	Generate sensor data	0.67	0.23	0.60
6	11 p.m.–7 a.m.	Public network	New device	Penang	Generate sensor data	0.11	0.34	0.60
7	11 p.m.–7 a.m.	House network	Allowed list	Other location	Generate sensor data	0.13	0.24	0.87



```
13:44:37.236 -> Humidity: 55.90 %      Temperature: 34.60 *C 94.28 *F  Heat index: 42.07 *C 107.73 *F
13:44:40.390 -> Humidity: 55.90 %      Temperature: 34.50 *C 94.10 *F  Heat index: 41.81 *C 107.26 *F
13:44:43.496 -> Humidity: 55.90 %      Temperature: 34.60 *C 94.28 *F  Heat index: 42.07 *C 107.73 *F
13:44:46.663 -> Humidity: 55.90 %      Temperature: 34.50 *C 94.10 *F  Heat index: 41.81 *C 107.26 *F
13:44:49.793 -> Humidity: 55.80 %      Temperature: 34.50 *C 94.10 *F  Heat index: 41.77 *C 107.18 *F
13:44:52.918 -> Humidity: 55.80 %      Temperature: 34.50 *C 94.10 *F  Heat index: 41.77 *C 107.18 *F
13:44:56.060 -> Humidity: 55.90 %      Temperature: 34.50 *C 94.10 *F  Heat index: 41.81 *C 107.26 *F
13:44:56.441 -> Attempting MQTT connection to 192.168.1.101 failed, rc=5 try again in 5 seconds
13:45:01.454 -> Attempting MQTT connection to 192.168.1.101 failed, rc=5 try again in 5 seconds
13:45:06.476 -> Attempting MQTT connection to 192.168.1.101 failed, rc=5 try again in 5 seconds
```

☑ Autoscroll ☑ Show timestamp Newline 9600 baud Clear output

Figure 12. Terminated sensor connection.

Based on Figure 12, the adaptive monitoring security agent identifies this security breach and checks for the existence of the user and IP address. The user with that IP address device info information is updated as a blacklisted device. The agent kicks out the active connection to prevent any data alteration or loss.

6. Discussion

In this section, discussions are presented in terms of enhanced MQTT authentication, contextual risk model, adaptive monitoring security agent, and an evaluation of the proposed architecture.

6.1. Enhanced MQTT Authentication Method in Fog-IoT Network

The MQTT protocol is used as the communication protocol for the Fog-IoT setup. The basic authentication method of MQTT protocol uses a username and password, and it is unencrypted. The authentication information can be hacked by dissecting the MQTT packet. Every existing MQTT broker allows a different authentication technique which depends on the scale of the project. The EMQX MQTT broker enhances the authentication method since the broker is scalable and highly extensible. The broker allows pre-trigger as a plugin for client authentication. The plugin is used to configure the command to check the username, IP address, and risk value against the MySQL database, which is shared with the access control mechanism. A unique token is introduced in the authentication secret between the sender and receiver in the communication channel. If a device's risk is more than the threshold risk value of 0.3, the device is disallowed to authenticate. This method is simulated in the Fog-IoT and evaluated. Hence, the first objective is achieved in this research.

6.2. Contextual Risk Model to Quantify Risk

A preliminary analysis is conducted on current risk models to address the Fog-IoT network challenges. A dynamic access control model is used to quantify the risk factor because it considers access policies and dynamic contextual features that can estimate in real-time [30]. This is apt in the Fog-IoT network since it is a distributed and flexible network. Two types of risk estimation methods are created to quantify the risk: risk assessment and the fuzzy model. Based on the existing models, the proposed model is enhanced to support adaptive features by adding the history and device context to better estimate risk. The risk assessment model is used with a Multifactor Evaluation Process to provide precise risk calculation while the Fuzzy model approaches enhanced with a risk assessment formula using the MATLAB tool. Both models are simulated in the Fog-IoT setup and were evaluated. The risk factor results are compared with a series of scenarios to obtain accurate results. The enhanced fuzzy model provided better results in seven different random scenarios testing with the same Fog-IoT setup using the risk assessment. Thus, the fuzzy model is used to quantify the risk, and the second objective of this study is achieved.

6.3. Adaptive Monitoring Security Agent

An adaptive monitoring security agent is developed in the fog node and an actual fog node that filters the clients' data by conceptualizing the Adaptative Security Architecture (ASA) model demonstrated in Figure 1; the whole solution of the adaptive security monitoring involved the stages of detection, prevention, responses and prediction. The agent constantly looks for unauthorised access or connection-related log information from the EMQX MQTT broker. The agent updates client info as a blacklisted device in the MySQL database if a particular client is unable to authenticate. The agent also terminates any connected client in the channel by rerunning the access control of the EMQX MQTT broker. The adaptive monitoring security agent is simulated in the normal authentication method with valid and invalid client info. The agent also tested the security analysis of the Man-in-the-Middle attack scenario. The adaptive monitoring agent is able to complement the enhanced MQTT authentication technique of this proposed overall architecture. Therefore, the third objective of this study is achieved.

6.4. Evaluation of Proposed Architecture

Each objective of this study is evaluated by running the simulation and series of experiments and enhanced the MQTT authentication method simulated by using NodeMCU with *dht22* sensors and localhost EMQX MQTT server. The simulation is run by tweaking the input parameters to get the desired result. Besides, the fuzzy model is designed using MATLAB tool and automation script implemented to obtain the input and generate the output. The computation of risk evaluated with random scenarios and the fuzzy set range selection is tweaked to obtain a better degree. Lastly, the security analysis is performed by running a Man-in-the-Middle attack to validate the behaviour of the proposed model. The adaptive features of this proposed model are able to terminate the suspicious connection and update the risk factor as well.

6.5. Benchmarking Analysis

The researcher conducted the experiments on existing models and benchmarked against the proposed model. The proposed model shows accurate results over seven different scenarios for the same action severity. The fuzzy model by Al-Zewairi et al. (2016) [31] almost obtained an accurate risk factor in case 1 and shows a vast difference in other conditions. However, Al-Zewairi et al.'s (2016) [31] model has no context data embedded in its model. Lakshimi et al.'s (2015) [32] proposed risk model takes no input from past risk. The main difference stated in our work is the combination of both context and past risk factors, which have huge differences in any risk-based authentication. These parameters are necessary to quantify risk factors because they can influence the value even with a small change in an environment.

Based on Table 6, few device authentication risk assessments are being displayed.

Table 6. Risk assessment benchmarking Results.

Cases/Model	Fuzzy Model by Al-Zewairi et al. (2016) [31]	Risk Assessment Model by Lakshmi et al. (2015) [32]	Proposed Model
1	0.38	0.27	0.41
2	0.30	0.27	0.14
3	0.60	0.27	0.36
4	0.38	0.27	0.15
5	0.60	0.27	0.60
6	0.13	0.27	0.60
7	0.13	0.27	0.87

For instances, any risk threshold higher than 0.3 is set to be a medium risk. Medium risk values recorded by Case 1, 3, 5, 6 and, 7 simply mean that, since the risk values are higher than accepted threshold during the authentication process, it would be flagged and therefore no further authentication process would take place. In addition, the result is stored in the database. In the future, in case a valid entry risk is low, it can be re-authenticated.

7. Conclusions and Future Work

There are three main contributions of this research which are enhanced by the basic MQTT protocol authentication method with a risk factor and token-based approach using the MYSQL database, proposed adaptive fuzzy-based access control model with device context and behavioural aspects which current access control models lacking off, and the proposed adaptive monitoring agent lightweight model which uses less resources to monitor the network in the fog server. For future work, the adaptive risk model proposed in this research can be enhanced further to support more parameters and factors which can influence the risk factor calculation. The authentication phase can be the primary focus because it is the first layer to intercept any network communication. In addition, the authentication process with risk computation can be a centralised entity that reduces the validation steps. The decision-making process can be instantaneous during a session whenever the device context changes into a different state. The session needs to be efficient and dynamic to provide the best service for a system. Since the risk computation and active monitoring can be centralised, resource management also can be optimised. The usage of nonrelational databases such as NoSQL and Mongo Db could provide this optimisation as well. This provides a robust security solution besides being able to cater the heterogenous nature of IoT data as well.

Based on the simulations performed and the results obtained, one major contribution towards the body of knowledge is that the usage of risk-based authentication for any IoT-edge environment is a mandatory requirement. However, an optimized risk results are possible when a fuzzy based risk model is adopted. The security properties in any IoT-Edge environment are beyond a risk-based authentication but also a secure IoT communication protocol such as MQTT. Thus, in real-implementation, our simulated environment using an IoT-edge proof-of concept could be adapted to secure any edge environment.

Author Contributions: Conceptualisation, S.S. and M.M.S.; methodology, S.S.; software, S.S.; validation, S.S.; formal analysis, S.S.; investigation, S.S.; resources, S.S.; data curation, S.S.; writing—original draft preparation, S.S.; writing—review and editing, M.M.S.; visualisation, S.S.; supervision, M.M.S.;

project administration, M.M.S.; funding acquisition, M.M.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Research Center Management Office USM.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data is contained within the article.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A

Table A1. Input and output variables for Fuzzy.

Input Variable: Time		
Value	Notation	Range
Low	L	[0 0 0.5]
Medium	M	[0 0.5 1]
High	H	[0.5 1 1]
Input Variable: Network Type		
Value	Notation	Range
Low	L	[0 0 1]
High	H	[0 1 1]
Input Variable: Device Info		
Value	Notation	Range
Low	L	[0 0 0.5]
Medium	M	[0 0.5 1]
High	H	[0.5 1 1]
Input Variable: Location		
Value	Notation	Range
Low	L	[0 0 1]
High	H	[0 1 1]
Input Variable: Action		
Value	Notation	Range
Low	L	[0 0 0.25 0.4]
Medium	M	[0.35 0.5 0.7]
High	H	[0.6 0.7 1 1]
Input Variable: Past Risk		
Value	Notation	Range
Low	L	[0 0 0.4]
Medium	M	[0.3 0.5 0.7]
High	H	[0.6 1 1]
Output Variable: Risk		
Value	Notation	Range
Negligible	N	[0 0.01455 0.179 0.3]
Low	L	[0.1 0.3 0.4]
Medium	M	[0.2 0.343 0.517 0.6]
High	H	[0.4 0.6 0.8]
Very high	VH	[0.743 0.793 0.941 1]

References

1. Al Agha, K.; Pujolle, G.; Ali-Yahiya, T. *Mobile and Wireless Networks*; John Wiley Sons: Hoboken, NJ, USA, 2016; Volume 2, pp. 283–306.
2. Grover, K.; Lim, A. A survey of broadcast authentication schemes for wireless networks. *Ad Hoc Netw.* **2015**, *24*, 288–316. [CrossRef]
3. Grammatikis, P.I.R.; Sarigiannidis, P.G.; Moscholios, I.D. Securing the Internet of Things: Challenges, threats and solutions. *Internet Things* **2019**, *5*, 41–70. [CrossRef]
4. Atlam, H.F.; Walters, R.J.; Wills, G.B. Fog Computing and the Internet of Things: A Review. *Big Data Cogn. Comput.* **2018**, *2*, 10. [CrossRef]
5. Gartner. Gartner Adaptive Security Architecture Model. Gartner Presentation. 2016. Available online: https://www.gigamon.com/lp/gartner-adaptive-security-architecture-model.html (accessed on 20 August 2021).
6. Wilson, Y.; Hingnikar, A. OpenID Connect. In *Solving Identity Management in Modern Applications*; Apress: New York, NY, USA, 2019; pp. 77–97. [CrossRef]
7. Mohanapriya, M.; Krishnamurthi, I. Modified DSR protocol for detection and removal of selective black hole attack in MANET. *Comput. Electr. Eng.* **2014**, *40*, 530–538. [CrossRef]
8. Aziz, B.A. On the security of the MQTT protocol. *Eng. Secur. Internet Things Syst.* **2016**, 159–178. [CrossRef]
9. Agoni, A.; Dlodlo, E.M. IP Spoofing Detection for Preventing DDoS Attack in Fog Computing. In Proceedings of the 6th Global Wireless Summit (GWS) 2018, Chiang Rai, Thailand, 25–28 November 2018; pp. 43–46. [CrossRef]
10. Stojmenovic, I.; Wen, S. The Fog computing paradigm: Scenarios and security issues. In Proceedings of the 2014 Federated Conference on Computer Science and Information Systems, Warsaw, Poland, 7–10 September 2014. [CrossRef]
11. Kartheek, D.N.; Bhushan, B. Security Issues in Fog Computing for Internet of Things. In *Architecture and Security Issues in Fog Computing Applications*; IGI Global: Hershey, PA, USA, 2019.
12. Diep, N.N.; Lee, S.; Lee, Y.K.; Lee, H. Contextual risk-based access control. In Proceedings of the International Conference on Security & Management, SAM 2007, Las Vegas, NV, USA, 25–28 June 2007; CSREA: Las Vegas, NV, USA; pp. 406–412.
13. Bonomi, F.; Milito, R.; Zhu, J.; Addepalli, S. Fog Computing and Its Role in the Internet of Things. In *Advancing Consumer-Centric Fog Computing Architectures*; Munir, K., Ed.; IGI Global: Hershey, PA, USA, 2012; pp. 63–71. [CrossRef]
14. Khan, S.; Parkinson, S.; Qin, Y. Fog computing security: A review of current applications and security solutions. *J. Cloud Comput.* **2017**, *6*, 1–22. [CrossRef]
15. Vaquero, L.M.; Rodero-Merino, L. Finding your Way in the Fog. *ACM SIGCOMM Comput. Commun. Rev.* **2014**, *44*, 27–32. [CrossRef]
16. Liu, J.; Xiao, Y.; Chen, C.L.P. Authentication and access control in the Internet of things. In Proceedings of the 2012 32nd International Conference on Distributed Computing Systems Workshops, Macau, China, 18–21 June 2012; pp. 588–592. [CrossRef]
17. Alrawais, A.; Alhothaily, A.; Hu, C.; Cheng, X. Fog. Computing for the Internet of Things: Security and Privacy Issues. *IEEE Internet Comput.* **2017**, *21*, 34–42. [CrossRef]
18. Cheng, P.C.; Rohatgi, P.; Keser, C.; Karger, P.A.; Wagner, G.M.; Reninger, A.S. Fuzzy Multi-Level Security: An experiment on quantified risk-adaptive access control. In Proceedings of the 2007 IEEE Symposium on Security and Privacy (SP '07), Berkeley, CA, USA, 20–23 May 2007; pp. 222–227. [CrossRef]
19. Li, J.; Bai, Y.; Zaman, N.A. fuzzy modeling approach for risk-based access control in eHealth cloud. In Proceedings of the 2013 12th IEEE International Conference on Trust, Security and Privacy in Computing and Communications, Melbourne, VIC, Australia, 16–18 July 2013; pp. 17–23. [CrossRef]
20. Ni, Q.; Bertino, E.; Lobo, J. Risk-based access control systems built on fuzzy inferences. In Proceedings of the 5th ACM Symposium on Information, Computer and Communications Security, Beijing, China, 13 April 2010; pp. 250–260. [CrossRef]
21. Khambhammettu, H.; Boulares, S.; Adi, K.; Logrippo, L. A framework for risk assessment in access control systems. *Comput. Secur.* **2013**, *39*, 86–103. [CrossRef]
22. Shaikh, R.A.; Adi, K.; Logrippo, L. Dynamic risk-based decision methods for access control systems. *Comput. Secur.* **2014**, *31*, 447–464. [CrossRef]
23. Rajbhandari, L.; Snekkenes, E.A. Using Game Theory to Analyse Risk to Privacy: An Initial Insight. *IFIP Adv. Inf. Commun. Technol.* **2010**, *352*, 41–51. [CrossRef]
24. Sharma, M.; Bai, Y.; Chung, S.; Dai, L. Using risk in access control for cloud-assisted ehealth. In Proceedings of the 2012 IEEE 14th International Conference on High Performance Computing and Communication & 2012 IEEE 9th International Conference on Embedded Software and Systems, Liverpool, UK, 25–7 June 2012; pp. 1047–1052. [CrossRef]
25. Diep, N.N.; Hung, L.X.; Zhung, Y.; Lee, S.; Lee, Y.K.; Lee, H. Enforcing access control using risk assessment. In Proceedings of the Fourth European Conference on Universal Multiservice Networks (ECUMN'07), Toulouse, France, 14–16 February 2007; pp. 419–424. [CrossRef]
26. Ficco, M.; Esposito, C.; Xiang, Y.; Palmieri, F. Pseudo-Dynamic Testing of Realistic Edge-Fog Cloud Ecosystems. *IEEE Commun. Mag.* **2017**, *55*, 98–104. [CrossRef]
27. Ahmadi, P.; Islam, K.; Maco, T.; Katam, M. A Survey on Internet of Things Security Issues and Applications. In Proceedings of the 2018 International Conference on Computational Science and Computational Intelligence (CSCI) 2018, Las Vegas, NV, USA, 12–14 December 2018. [CrossRef]

28. Xu, X. Study on security problems and key technologies of the internet of things. In Proceedings of the 2013 International Conference on Computational and Information Sciences, Shiyang, China, 21–23 June 2013; pp. 407–410. [CrossRef]
29. Babar, S.; Stango, A.; Prasad, N.; Sen, J.; Prasad, R. Proposed embedded security framework for Internet Things (IoT). In Proceedings of the International Conference on Wireless Communication, Vehicular Technology, Information Theory and Aerospace and Electronic Systems Technology (Wireless VITAE), Chennai, India, 28 February–3 March 2011. [CrossRef]
30. Dos Santos, D.R.; Westphall, C.M.; Westphall, C.B. A dynamic risk-based access control architecture for cloud computing. In Proceedings of the 2014 IEEE Network Operations and Management Symposium (NOMS), Krakow, Poland, 5–9 May 2014. [CrossRef]
31. Al-Zewairi, M.; Suleiman, D.; Shaout, A. Multilevel Fuzzy Inference System for Risk Adaptive Hybrid RFID Access Control System. In Proceedings of the 2016 Cybersecurity and Cyberforensics Conference (CCC), Amman, Jordan, 2–4 August 2016; pp. 1–7. [CrossRef]
32. Lakshmi, H.; Namitha, S.; Seemanthini; Gopalan, S.; Sanjay, H.A.; Chandrashekaran, K.; Bhaskar, A. Risk based access control in cloud computing. In Proceedings of the 2015 International Conference on Green Computing and Internet of Things (ICGCIoT), Greater Noida, India, 8–10 October 2015. [CrossRef]