ACTA SCIENTIFIC COMPUTER SCIENCES

Volume 2 Issue 7 July 2020

Review Article

# AMIS: Authentication Mechanism for IoT Security

**Rachid Zagrouba***

*College of Computer Science and Information Technology, Imam Abdulrahman Bin Faisal University, Dammam, Saudi Arabia*

***Corresponding Author:** Rachid Zagrouba, College of Computer Science and Information Technology, Imam Abdulrahman Bin Faisal University, Dammam, Saudi Arabia.

**Received:** June 23, 2020

**Published:** June 30, 2020



## Abstract

The Internet of Things (IoT) is composed of different networked objects, for example, the smart devices which are interconnected to gather, process, refine and exchange meaningful data over the Internet. In this context, this research paper is devoted to the study of different algorithms and techniques of authenticating internet of things devices. This paper introduces the internet of things field, the background talked about both internet of things devices and the authentication that can be applied in these devices. Additionally, discussed some algorithms and techniques of authenticating the devices of internet of things like the principle of lightweight identity based elliptic curve cryptography (ECC) and Lamport's one-time password (OTP) algorithm, lightweight cryptographic in IoT ecosystem, a protocol used symmetric-key cryptography and hash message authentication code based key derivation function, a lightweight continuous authentication protocol, a Mutual Authentication in IoT Systems Using Physical Unclonable Functions and proximity-based mechanism for IoT devices authentication. Moreover, this paper compares and analyses the mentioned approaches to select the best solution. The result of the comparison and analysis shows that the best solution is Mutual Authentication in IoT Systems Using Physical Unclonable Functions.

**Keywords:** IoT; Authentication; Cybersecurity; Countermeasure; Attacks

## Introduction

In the golden age of technology, the internet of things concept has increased significantly. A huge number of IoT devices are used, it has been estimated that the number of using IoT devices will reach to 14.6 billion by 2022 [1]. This technology made a great difference in human daily life such as in industrial and healthcare aspects.

Although there are a lot of features in this technology, there is a huge problem that the IoT devices face which is security. Due to the sensitive information that is transmitted and stored in the IoT devices, it is exposed to several security threats. So, any compromise happens to these devices, will affect the user. Authenticate the device in internet of things environment is very important to ensure that legitimate user is communicating with target devices, not someone else. The problem is that the authentication mechanism needs to be protected against attacks and make efficient use of the limited resources by consuming low battery power and make use of the small storage. The conventional cryptography algorithms such as RSA, ECC algorithms are not appropriate for IoT devices because these devices have limited resources in term of battery life, computational power, storage space and bandwidth [2]. Despite, there are many algorithms and techniques that are used in order to authenticate the IoT devices, not all of them is sufficient to protect against attacks and consider the limited resources. Thus, the mechanism used in such devices should consider the restricted resources during the authentication process. Discussing these algorithms and techniques assist to understand and know how the IoT devices are authenticated and how the algorithms and techniques are robust. This paper will discuss the strengths and weaknesses of each authentication algorithm and technique in order to enhance the IoT security.

This paper organized as follows: Background about the discuses topic. After that, it discusses the literature review of existing solutions. In addition, the reviewed papers are compared and analysed. The last section will be reserved to the presentation of the proposed model to improve the IoT security.

## Literature Review

This section presents published papers related to different existing solutions to authenticate IoT devices to propose a solution that enhances the security field of the internet of things environment.

Thanushree., *et al*. proposed in [6] scheme used the principle of lightweight identity based elliptic curve cryptography (ECC) and

Lamport's one-time password (OTP) algorithm. Each of end-to-end authentication consists of the device are ECC and Lamport's algorithm to provide authentication and secure communication for IoT devices. The ECC consists of three steps: Key generation, encryption and decryption. When the client requests a server and the server decides to connect to this client, the server then will start the ECC which generates a pair of keys for every session that maintaining the security. The pair of keys are private and a public key that should be present in client and server for using those keys for encryption and decryption. So, the authors proposed Lamport algorithm as two-factor authentication. Lamport's OTP is a sequence of a random number-based factor from the server that gives a specific number from 1 to 1000 or 1 to 100... etc. Lamport function in the client side gives the next series number whereas the Lamport inverse function in the server side gives the previous number. The client will generate the OTP and it sends along encrypted the message to the server. The server will be checking whether the current OTP received is the same as the previous OTP received or not. If the current OTP equal the previous OTP then the client is authenticated, if not then the client is unauthenticated. The proposed schema has advantages: first, it uses small key size, second it does not require additional infrastructure is the need which leads to reduce the possibility of attack.

In June 2017, the authors discussed the importance of the lightweight cryptographic in IoT ecosystem [7]. This paper tries to utilize the limited space of memory, reducing the consumption of the power in the smart devices that have limited resources. Consequently, the authors proposed a novel approach that combines the two algorithms which are symmetric and asymmetric algorithms. The proposed algorithm called "hybrid lightweight algorithm". This approach gives each smart device the lightweight cryptographic algorithm that it is suitable for its abilities. The parameters used to determine the proper choices are "data size, battery power, memory space, and computation power". After the values of the mentioned parameters are provided, each parameter will be compared with a threshold value. The Threshold value is calculated using a specific algorithm. It has four phases each phase evaluates one parameter, after evaluating all the parameters, the smart devices will be provided with the proper algorithm. However, this approach does not protect from possible attacks such as modules attacks, timing attack, and man-in-the-middle attack.

Rabiah., *et al*. proposed in [8] a protocol used symmetric-key cryptography and hash message authentication code based key derivation function. The protocol provides authentication, key exchange, confidentiality, and message integrity. Moreover, the protocol depends on both devices having two unique keys which are a shared long-term symmetric master key ($K_m$) and shared short-term symmetric session key ($K_s$). Two unique keys stored in non-volatile memory. Each session has a maximum number of messages to be exchanged in the session. The message will be exchanged in the session is encrypted by ($K_m$) and hashed by an initial session key ($K_{iks}$). In addition, $K_s$ is used for encrypting and hashing session message, and it will be updated based on a random set of the frame exchanged in the previous session $K_s$ and the master key $k_m$. If the attacker wants to launch a man in the middle attack and he knows $k_m$ only, then he cannot succeed this attack because he must know the $k_{iks}$. As the $k_s$ it depends on the previous $k_s$ thus the attacker would not calculate the current $k_s$. Furthermore, the feature in this proposed protocol does not depend on any trusted third party to operate IoT devices in disconnected.

In April 2018, Chuang., *et al*. in [9] proposed a protocol for IoT authentication. It called "lightweight continuous authentication protocol" which consists of three phases the three phases are initialization phase, static authentication, continuous authentication phases. The first phase is the initialization phase, this phase set up the essential parameter in the gateway and sensor which will be transformed into a secure channel. The sensor sends to the gateway its identity IDSN and other information like the lifetime and capacity of the battery. When the information sends by the sensor and delivered to the gateway, the gateway will calculate the average battery consumption value of the sensor. After that, the gateway calculates the secret value, the gateway will send calculated secret value to the sensor to store it in its storage. In addition, the gateway will determine the authentication period to accelerates the transmission of authenticating data. The second phase is static authentication which is similar to the conventional authentication method. In this phase, the two devices are mutually authenticating each other. It generates an authentication token that is used in the next phase which is continuous authentication phase. The gateway computes the threshold capacity of the remaining battery in order to guarantee that the remaining value is enough to send data to the gateway. In static authentication phase, the sensor sends six parameters to the gateway which are Identity of the sensor ($ID_{SN}$), Messge1 ($M_1$), Messge2 ($M_2$), Masked battery capacity of the node (mb), a random number ($r_1$) and X. These values are computed and sent to the gateway. Thence, the gateway verifies the correctness of the send data by computing each parameter. the last phase is continuous authentication that transmits data from the sensor to the gateway before the determined period of the time is finished. The gateway verifies the node's authenticity after data transmission. The gateway ensures whether or not the message that sends by the node is generated during the determined period of time as well as check the value of message 5 that shows the integrity of the send message and the amount of the remaining battery. The proposed protocol has various features such as protection against

man the middle, impersonate, replay attacks. Moreover, there are additional features exist in the proposed protocol like mutual authentication and data integrity.

The proposed solution in [10] present light-weight mutual authentication protocols for IoT systems based on physical unclonable functions. There are two scenarios are presented for mutual authentication. An IoT device wants to establish a connection with a server in the datacentre. The second one is when one IoT device wants to talk to another IoT device. For this purpose, this paper tries to use Physical Unclonable Functions (PUFs). The chip manufacturing process announced random differences in the PUFs at the physical sub-microscopic structure in the form of a noisy function that is implanted into a physical circuit. Therefore, this means that it is not possible to create two PUFs to be exactly the same and at the same time there is no need for any secrets on the IoT device and also make sure the IoT device cannot be cloned. The paper relays on the secret keys or passwords which is the PUFs and they are used as secret keys or passwords to set a hardware fingerprint to IoT devices. Furthermore, there is a CPU and radio energy consumed by the proposed protocol like the other existing schemes but in a way that it has lower CPU and radio transceiver energy consumption. However, to use the proposed protocol for applications with strict timing requirements, such as vehicular networks, it is desirable to further reduce the latency of authentication by reducing the number of messages exchanged between the entities.

In [11] a proximity-based mechanism for IoT device authentication is proposed and it is called Move2Auth for enhancing IoT device security. In Move2Auth, the user will do such hand-gestures like moving towards and away or rotating in front of the IoT device while holding the smartphone as shown in figure 1. Move2Auth can consistently detect proximity and authenticate IoT device by joining the large RSS-variation and the matching between RSS-trace and smartphone sensor-trace. This technique can be used against impersonating attack and eavesdroppers. As IoT devices are mostly embedded devices, there is a big challenge in the authentication due to the lack of PSK. In the same context, the authors in [11] did not test the technique in all IoT types and assumed that the IoT device does not contain a screen or keyboard or any sophisticated user interface and does not have sensors like a camera and microphone. Additionally, the IoT device is not easy to move like a power switch plugged on walls. Many IoT device manufacturers connect the IoT device to the smartphone first by forcing smartphone to input PSK. In [11] IoT device can gain PSK from smartphone and do PSK based authentication with the router if the connection between smartphone and IoT device is secure. In this way, the problem is reduced from router-IoT authentication to smartphone-IoT authentication.

**Figure 1:** Proximity based IoT device authentication.

## Comparison and analysis

### Comparison

The solutions mentioned in the reviewed papers are compared in this section in order to analyse them and determine which one provides an effective solution. The comparison used five criteria to evaluate each mechanism. According to the limitations in the internet of things devices such as computational power, memory, and energy. So, the criteria are the computational efficiency, communication overhead, mutual authentication, and resistance against attacks.

### Analysis

This section provides an analysis for the compared solutions in table 1 and 2 to determine the best solution.

| Criteria | Description |
|---|---|
| Computational efficiency | Is the property of a technique or algorithm in how to use computational resources such as CPU, and memory in an efficient way. |
| Communication overhead | The parameters size of the message transmitted between two entities during the authentication [12]. |
| Mutual authentication | The two-parties authenticate each other before starting the communication [13]. |
| Resistance against attacks | The strength of the mechanism to protect the devices from different attacks. |

**Table 1:** Definition of the evaluated criteria.

The table 2 shows that the result of the proposed schema in [6] indicates that it has low communication overhead which means that the size of parameters is small. As for computational efficiency, this proposed solution only required less usage of memory as well as less CPU consumption. The proposed solution in [6] confirms that the authentication process is done from server-side only. Thus, the algorithm is not achieving mutual authentication.

**Table 2:** Comparison between mechanism.

Although the mechanism proposed by Singh., *et al*. in [7] achieved computational efficiency because it gives each device the algorithm that is suitable with its abilities, it is vulnerable to two types of attacks which are man-in-the-middle.

The proposed protocol in [8] exchanges the limited number of messages in every session because the smart devices have small memory space. Moreover, the symmetric key cryptography uses a smaller key as well as need smaller space in the memory compared to existing public key cryptography. So, this protocol achieves computational efficiency. However, the proposed protocol does not achieve low communication overhead. Additionally, it has mutual authentication because both devices authenticate each other in the initial phase using $k_m$ and $k_{iks}$. Moreover, as shown in the table 4 the protocol is robust against the man-in-the-middle attack, replay attack, impersonate attacks.

In [9], the authors present another solution, where mutual authentication and computational efficiency are accomplished. Mutual authentication accomplished when the two devices authenticate each other using M2, M4 with their secret value. The proposed solution in this paper protects against replay and impersonated attacks, where the attacker does not have the random number that is generated and transmitted with the message. Thus, the receiver will know that the message is fake when compares the received message with the computed message. For the impersonate attack, it will be difficult for the attacker to impersonate as a legitimate device because he cannot forge M2 except he has the secret key of the sensor which is difficult to get. Additionally, this solution has immunity against man-in-the-middle attack as shown in table 4. The attacker will not be able to manipulate the message except if he knows the secret value and the remaining battery of the device which cannot be known from the previous eavesdropping. Thus, this solution can prevent such attacks.

| Criteria | Description |
|---|---|
| Replay attack | It is a type of attacks where the transmitted data is sniffed by the illegitimate user in order to transmit the repeated or delay data [14]. |
| Impersonate attack | It is a type of attacks that deceives the other party by impersonating as a legitimate party in a system [15]. |
| Man-in-the middle attack | The attacker intercepts a conversation between two parties. This conversation appears to be between the two parties directly. However, it is controlled by the attacker that stands in the middle of the channel, who can view, add, remove, modify and replace the messages exchanged in this conversation [16]. |
| Physical attack | Physically access to the devices and stealing the secret keys where its lead to clone the device. |

**Table 3:** Definition of attack.

**Table 4:** Comparison based resistance against attacks.

The technique proposed in [10] which discusses the Mutual Authentication in IoT Systems Using Physical Unclonable Functions has a computational efficiency based on specific statistics done by the researchers and low communication overhead in the use of hardware fingerprint to authenticate the IoT device. Furthermore, there is mutual authentication and the IoT device can authenticate the user also. However, the location of an IoT device using the wireless channel characteristics to generate wireless fingerprint is not achieved. In addition, this technique is resistant against attacks like replay attack, impersonate attack and man-in-the-middle attack.

In the same context, Zhang., *et al*. in [11] in the proposed proximity-based mechanism for IoT device, authentication has the same result of [10] in computational efficiency, low communication overhead and resistance against attacks. However, there is no mutual authentication in the proximity-based mechanism for IoT device authentication and the IoT device cannot authenticate the user as

shown in table 2. Moreover, this technique is resistant against the replay attack, impersonate attack, man-in-the-middle attack.

The analysis showed that all reviewed papers have computational efficiency because all of them using lightweight authentication mechanisms for smart devices. In addition [6,10,11] have low communication overhead. However, [7-9] does not provide any information about low communication overhead feature. Where [7-10] have mutual authentication unlike [6,11]. Furthermore, all mechanisms have immunity against replay and impersonate attack expect [6] does not give any information. Paper [7] is vulnerable to man-in-the-middle and side-channel attack. As for [8] and [10] have protection against one more attack which is session hijacking attack in [8] and side-channel attack in [10] respectively. According to the conduct analysis, the best solution is the Mutual Authentication in IoT Systems Using Physical Unclonable Functions. To conclude, there is no mechanism that achieves all the mentioned features. Thus, there is a need to propose a new mechanism that provides all features.

## Proposed model

This section discusses the scheme proposed to enhance the Authentication Mechanism for IoT Security (AMIS). The proposed solution is based on adding additional factor of authentication that depends on the location of the IoT device.

Based on our comparison [11] is the best solution among the other solution for making the IoT authentication mechanism secure. Based on the comparison and analysis, this mechanism satisfies all the criteria of the table 4 it has computational efficiency, low communication overhead where the length of the message exchange during the authentication phases is short. In addition, it has mutual authentication where the server and IoT devices authenticate each other. The protection from MIM, reply, cloning, tampering and physical attack give the such mechanism robustness. The AMIS mechanism enhances by adding another factor of authentication to make the authentication more secure and prevent from other attack. Beside the PUF which is consider as hardware fingerprint that cannot be clone or even modify by anyone otherwise it will be destroy, the wireless fingerprint is added as second factor authentication. This wireless fingerprint has two Indicators that used to authenticate the IoT devices' two Indicators are received signal strength indicator (RSSI)and the link quality indicator (LQI). The RSSI measure the power of the received radio signal while LQI indicator of the link quality between a transmitter and receiver. By using wireless fingerprint the spoofing attack can be prevented where the IoT device should be exist in the known location in order to accept the authentication request coming from the device. In case the IoT device is stolen and the attacker try to impersonated as legal user and try to authenticate the IoT device from remote and unknow location the authentication will be rejected [19]. Figure 2 show mutual authentication process after the combination of the PUF with wireless fingerprint. The authentication process consists of 3 following steps.

1. The IoT device start the mutual authentication by sending $PID_A$ and $N_1$ to the server. The server in turn looking for CRP of $PID_A$ in its memory.

**Figure 2:** Proximity based IoT device authentication.

2. The server will reject the authentication if he founds the CRP. Otherwise, the server will read the CRP of IoT devices and create a $Rs_1$. Encrypted message using $R^i$ consist of the $Rs_1ID_A$, $N_1$. The sever send three parameter to the IoT devices which are $M_A$, $C^i$ and the MAC.
3. The $R^i$ is calculated using the $C^i$ and the PUF. Ri used to get $Rs_1$ and verify the MAC to continue the process of the authentication. After that, another Ri will be calculated using new $C^i$ and PUF for future authentication. The IoT devices will send $M_s$ and MAC before removing all the temporary parameters. Samples RSSI values and LQI of the wireless gateway $ID_G$ for the current packet is calculated in the server side.
4. The server will Ri used to calculate the $N_A$ and $R^i+1$ as well the Mac will be verified. The $P_r$ and var|LQI| which are the values of RSSI and LQI that sent by the wireless gateway calculated using the threshold values which are $_G$ the RSSI and $_G$ for the |LQI|. The threshold values is calculated based on specific formulas for each of them. If the values of the $P_r$ greater or equal $_G$ and the var|LQI| less or equal then $_G$ the authentication is accepted the. By this the two entities authenticate each other.

| Notation | Description |
|---|---|
| $ID_A$ | The identity of the IoT device |
| $R^i$ | The response of the PUF |
| $N_1$ | The nonce generate by the IoT device |
| $C^i$ | The challenge of the PUF |
| $Rs_1$ | Nonce generate by the server |

**Table 5:** Definition of notation.

## Conclusion and Future Work

In conclusion, this paper aims to prove a review, comparison, and analysis of different algorithms and techniques related to the authentication in IoT in the current state of security like Proximity based IoT device authentication and hybrid lightweight algorithm. We Proposed a new Authentication mechanism to improve the IoT security based on Mutual Authentication in IoT Systems Using Physical Unclonable Functions, by focusing on detecting the location of an IoT device using the wireless channel characteristics to generate wireless fingerprint and protect against more attack. For the future work, the proposed solution will be implemented to develop the performance evaluation with a real experimentation.

## Bibliography

1. Gagliordi N. "IoT to drive growth in connected devices through 2022". Cisco | ZDNet. [online] ZDNet (2020).

2. Shahzad M and Singh M. "Continuous Authentication and Authorization for the Internet of Things". *IEEE Internet Computing* 21.2 (2017): 86-90.

3. Mc Clelland C. IoT Explained - How Does an IoT System Actually Work (2020).

4. Vasee VV and IoT H. "Heart Rate Monitor Using IoT 97" (2018).

5. E Silva., *et al.* "Authentication and the Internet of Things: A Survey Based on a Systematic Mapping". In The Twelfth International Conference on Software Engineering Advances (2017).

6. Trtrastrnm and G Rao. "Integrating Lamport's One Time Password Authentication Scheme with Elliptic Curve Cryptography". *International Journal of Recent Trends in Engineering and Research* 2.4 (2016): 1-8.

7. Singh S., *et al.* "Advanced lightweight encryption algorithms for IoT devices: survey, challenges and solutions" (2019).

8. A Rabiah., *et al*. "A Lightweight Authentication and Key Exchange Protocol for IoT". Workshop on Decentralized IoT Security and Standards (2018): 1-6.

9. Chuang Y., *et al.* "A Lightweight Continuous Authentication Protocol for the Internet of Things". *Sensors* 18.4 (2018): 1104.

10. MN Aman., *et al*. "Mutual Authentication in IoT Systems Using Physical Unclonable Functions". *IEEE Internet of Things Journal* 4.5 (2017): 1327-1340.

11. J Zhang., *et al*. "Proximity based IoT device authentication". IEEE INFOCOM 2017 - IEEE Conference on Computer Communications (2017).

12. M Aman., *et al.* "A Light-Weight Mutual Authentication Protocol for IoT Systems" (2017).

13. Park N and Kang N. "Mutual Authentication Scheme in Secure Internet of Things Technology for Comfortable Lifestyle". *Sensors* 16.1 (2015): 20.

14. H Tilborg and S Jajodia. "Encyclopedia of Cryptography and Security, 2nd edition". Holland (2005): 1042.

15. "The Mechanics of Impersonation Attacks | The Email Laundry". Theemaillaundry (2017).

16. W Stallings. "Cryptography and Network Security: Principles and Practices". 7th edition, Prentice Hall, (2017).

17. T Hornby. "Side-Channel Attacks on Everyday Applications: Distinguishing Inputs with FLUSH+RELOAD (2014).

18. M Rouse. "What is session hijacking (TCP session hijacking)? - Definition from WhatIs.com". Search Software Quality (2006).

19. Ieeexplore-ieee-org.library.iau.edu.sa. Login e-Resources Portal | IAU (2019).