BITS 3533 Wireless Network and Mobile Computing

Sem 1 2022/2023

Assignment 1

Lightweight Wireless Protocol for IoT Application

Group 4 Members:

Muhammad Izham Bin Norhamadi	B032020039
Ahmad Sha Herizam Bin Tahir	B032020009
Affendy Elyas bin Azhari Sharidan	B032020024
Muhammad Rifqi Bin Ramlan	B032020028
Muhammad Ikmal Bin Mazlan	B032020002
Muhammad Khalif Asyara Bin Noor Shansuddin	B032110015

Lecturer:

TS. DR. NORHARYATI BINTI HARUM

TEAM MEMBERS AND ROLES

Group Member	In charge of
Izham	- Chapter 1: Introduction - 2.1 MQTT with Cryptographic Smart Card - 2.5 A Proposal for Bridging the MQTT Protocol to HTTP on IoT Solutions - 3.1 Comparison between MQTT and Http as an IoT protocol
Ahmad	- 2.2 IoT based Water Management System using MQTT protocol - 2.3 Blend of Cloud and Internet of Things (IoT) in agriculture sector using lightweight protocol - 2.6 A Lightweight Multi-tier S-MQTT Framework to Secure Communication between low-end IoT Nodes - 2.7 MQTT protocol employing IOT based home safety system with ABE encryption
Affendy	- A Lightweight Authentication Protocol for Internet of Things
Rifqi	- A Lightweight IoT Security Protocol
Ikmal	- 2.8 WBAN - 5.0 Conclusion
Khalif	- LAKA: Lightweight Authentication and Key Agreement Protocol for Internet of Things Based Wireless Body Area Network - 2.8 WBAN - 4.2 WBAN open issue and recommendation

TABLE OF CONTENT

CHAPTER 1: INTRODUCTION

The Internet of Things (IoT) is an ecosystem that provides the possibility of communications on internet to countless devices of various types such as sensors, devices, gateways, servers, appliances, user applications and many more. It is expected that by the end of 2022 there will be 20.4 billion of IoT devices connected to internet (V. Hassija, 2019). End Application requirements of range, data, security, power, and battery life dictate the network protocol of choice. One of the most appropriate communications protocols for the IoT is the Message Queuing Telemetry Transport (MQTT) protocol, due to its capacity for easy implementation on lightweight, cheap, low-power, and low memory devices (A. Al-Fuqaha, 2015) which allows communication between nodes in both reliable and unreliable networks.

MQTT protocol was designed by IBM and was standardized by OASIS (Open Architecture System) in 2013 and it was approved as ISO standard called ISO/IEC 20922 from 2016. MQTT follows a publish/subscribe architecture, meaning there are nodes (Brokers) that make the information available while clients can read the available information after subscribing by accessing corresponding URL. MQTT has three types of participants: the Broker, the Publisher, and the Subscriber. The Broker is the center of a star topology in MQTT protocol and oversees the exchange of messages between other participants. Every other participant can only connect with it as it also authenticates them in the network. The Publisher are the elements that send data to the Broker so that it sends this data to one or more Subscribers that require it. The Subscribers are the elements that receive data to the Broker which are sent by the Publishers.

When Kevin Ashton was creating the foundation for the Internet of Things (IoT) at MIT's AutoID lab in 1999, he invented the term "IoT." The internet of things is quickly evolving and becoming widely used because to improvements in wireless networking technology and more standardisation of communications protocols. All everyday items could talk with one another and be controlled by computers if they were given unique identifiers and wireless connectivity. As the Internet became more commercialised, security worries grew to include issues with financial transactions, individual privacy, and the risk of cybercrime. IoT security and safety go hand in hand. The IoT has significant hurdles in terms of security and privacy.

CHAPTER 2: TECHNICAL DESCRIPTION / DESIGN

2.1 Message Queuing Telemetry Transport (MQTT) with Cryptographic Smart Card

The implementation of Cryptographic Smart Card provides hardware secure, trustworthy, well tested and with low economic cost in the IoT devices to execute all necessary cryptographic functions, and a public key repository accessible for the broker (Sanjuan, 2020). With these new elements a new method for mutual authentication was presented in the MQTT protocol. An encryption schema was defined for encoding the data exchange between the clients and the broker, in both directions without including modifications in the specification of the protocol messages.

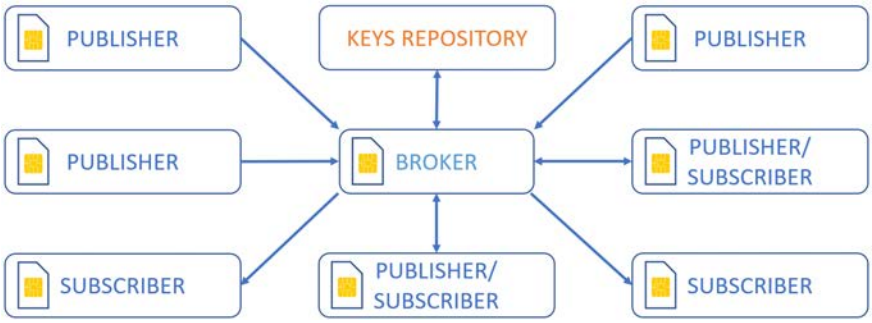


Figure 1 Publish/subscribe model of MQTT protocol with security schema based on cryptographic smart card

2.1.1 General Scheme of Security

To achieve the objectives in the article, there are three new elements in the system.

A. PUBLISHER/SUBSCRIBER CRYPTOGRAPHIC SMART CARD

The publishers and the subscribers MQTT must have a Cryptographic Smart Card that must be communicated with a microcontroller that manages the communication with the broker.

B. BROKER CRYPTOGRAPHIC SYSTEM

The broker must have a cryptographic system, either a Cryptographic Smart Card or an HSM (Hardware Security Manager) system and must complete several cryptographic functions for all messages sent between publishers and subscribers. If the number of clients and the frequency of messages produces messages overlaps, the system should execute several cryptographic operations in parallel with high speed.

C. PUBLIC KEY REPOSITORY

The system must have a public key repository accessible, through a secure protocol, by the broker. The implementer of this system can select the asymmetric cryptography algorithm (RSA_NOPAD, RSA_PKCS1, ECC, etc.), that he prefers for authentication proposes and the block cipher algorithm (AES, DES, TEA, NOEKEON, etc.) for payload encryption.

2.2 IoT based Water Management System using MQTT protocol

Nowadays, increasing environmental pollution, world population, water scarcity, etc. are some major issues faced by the society. We need solutions, which would be implemented from a individual person to large industries (Gaikwad, 2021). The ultrasonic sensor is used to measure the level of tank, the flow sensor is used to measure the flow rate of water and total consumed volume. This system is used to monitor usage of water, avoid overflow and saves water. All the information data such as level, flow rate and volume data will be transmitted by Node MCU to mobile app using the Message Queuing Telemetry Transport (MQTT) protocol.

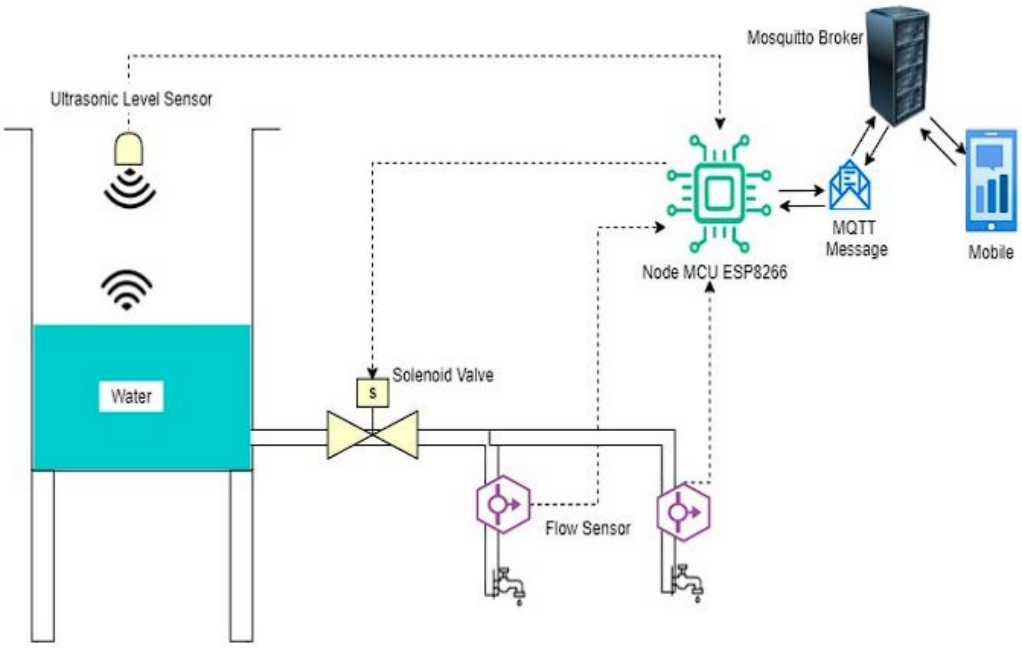


Figure 2 Water Management System using MQTT protocol

2.2.1 Technical Description

Firstly, Node MCU connected to internet through WiFi to calculates the volume, flow rate and level. After that, it sends the message to mosquitto broker under topic of ‘esp/flow’, ‘esp/flow’ and ‘esp/level’ respectively. Simultaneously it subscribe to topic ‘esp/valve’ to receive message to control valves. In any case of Node MCU failed to connect to mosquitto broker it will return code.

On mobile applications side, it will connect to MQTT mosquitto broker then subscribe to the same topic in Node MCU. Automatically, this wll make the app receive message from broker sent by Node MCU. All the messages will be stores in SQLite database and can be displayed in a graph.

2.3 Blend of Cloud and Internet of Things (IoT) in agriculture sector using lightweight protocol

A smart irrigation system is developed using the lightweight protocol, MQTT (Message Queue Telemetry Transport) since this protocol have 22% more energy efficient plus 15% faster compared to others protocol. All the information such as temperature and soil moisture data are collected and managed by Amazon Cloud. The data analysis is performed using the Weka (Waikato Environment for Knowledge Analysis) tool. With the invent of IoT and Cloud Computing, the agriculture sector has opened up a new door for “Precision agriculture”. Hence, the CloudIoT based irrigation system is designed and developed to conserve more water. Smart irrigation is capable of supplying the water to the entire field uniformly, schedule and the water supply remotely so that each plant has the adequate amount of water it needs (Raikar, 2018).

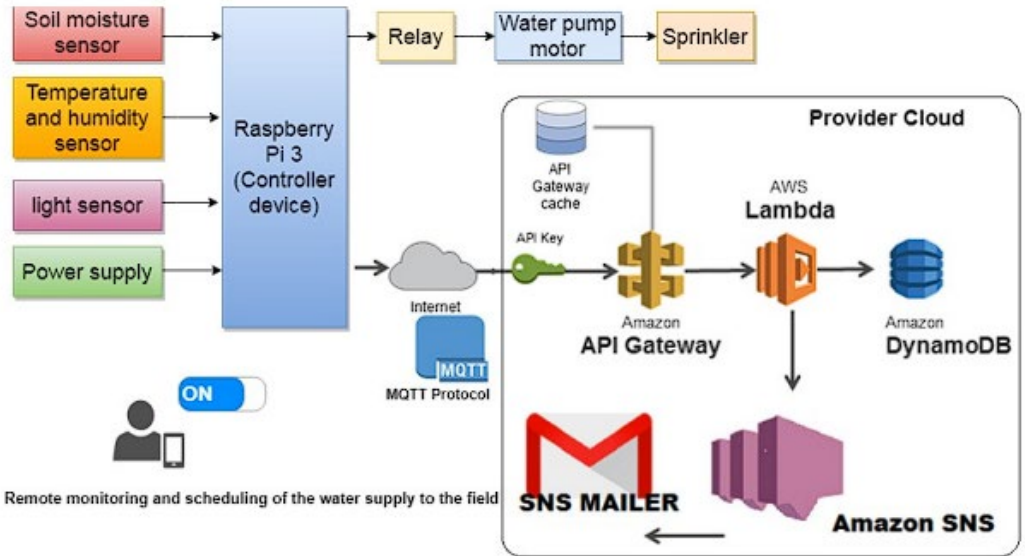


Figure 3 CloudIoT smart irrigation system

The CloudIoT architecture layers are applied to the case study smart irrigation system in the agriculture sector and divided in some parts:

- User Layer
  The usage of mobile app to turn on/off water motor remotely by user to increase water efficiency. User also can schedule the watering plants activity based on the soil moisture in the field.

- Proximity network
  IoT device: The temperature, moisture, and light intensity data are sent to the IoT gateway device
  IoT gateway device: Using the Raspberry Pi 3 as gateway device and connect it to AWS provider cloud

- Public Network
  IoT connectivity: MQTT is used for the communication between the client nodes and the service provider. MQTT protocol uses the ‘publish-subscribe’ type of communication model.

- Provider cloud
  Device management: MQTT server being executed using AWS IoT.
  API management: The access control keys are obtained for reading and writing data and also to implement AWS alarms and cloud watch services.
  Device data store: All sensor values is stored in the database for further referral.
  Data analysis: The Weka (Waikato Environment for Knowledge Analysis) tool used to perform data analysis such as classification and clustering,

2.4 A Lightweight Authentication Protocol for Internet of Things

A communication scenario used to creating the IOT using Radio Frequency Identification (RFID) tags via the Internet is shown on Figure 4

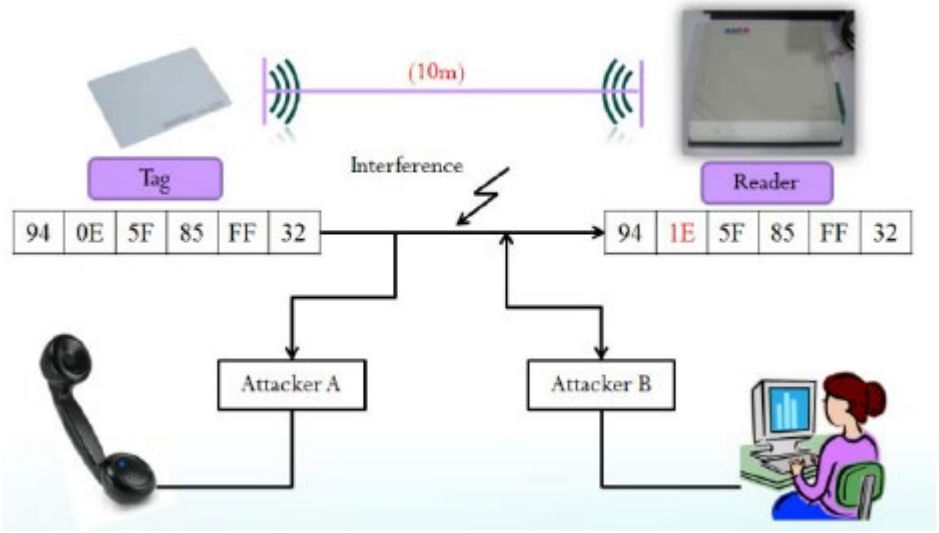


Figure 4 A communication scenario used to creating the IOT using RFID tag

The RFID readers are connected to the Internet. The tagged items are mobile and are expected to move through different reader fields and "connect" to the readers via their standard RFID communication protocol. The RFID reader identifies the tag using an appropriate authentication protocol. A real hurdle to designing secure RFID is the absence of cryptography in the technology. The Class-1 Gen-2 type of EPC tags have absolutely no specific anti-counterfeiting characteristics. In theory, a hacker could easily copy the EPC from a target tag and programmed it into a fake tag or mimic the target tag on a different kind of wireless device.

2.5 A Proposal for Bridging the MQTT Protocol to HTTP on IoT Solutions

Software that handles and stores IoT devices data is known as IoT middleware. The devices transfer data to middleware through an application protocol, which can be different from those supported by middleware. To overlap this problem, this paper proposes an application layer gateway for IoT protocols that can "translate" MQTT messages into HTTP reducing the packet size sent by an IoT device and it is fully configurable through a graphical user interface in runtime (da Cruz, 2018). The software responsible for such conversion is called application layer bridge (ALB), application layer gateway (ALG), or simply bridge.

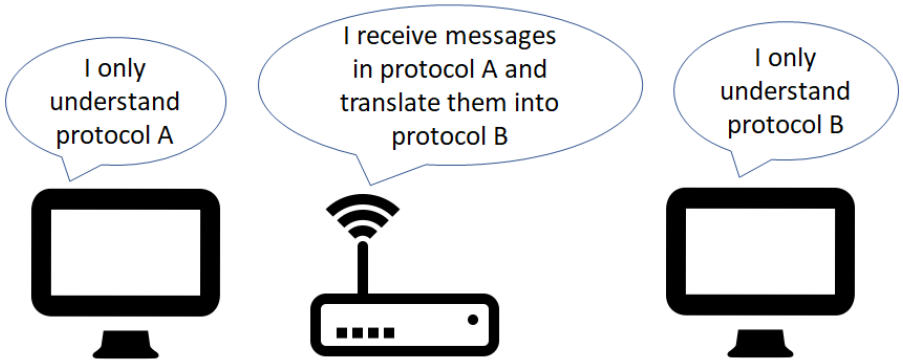


Figure 5 Illustration of a bridge operation where a gateway messages through a protocol A translates them to a protocol B

2.5.3 Bridging from MQTT to HTTP

The application layer gateway presented in this paper translates MQTT requests into HTTP. Its primary goal is to allow constrained devices to send fewer data when communicating with IoT middleware. The solution can be deployed on any device with Java installed which are widely used programming language. To use the solution, users must specify the MQTT broker details (IP, port, topic) and the middleware server details (IP, port, path). The software then subscribes to the specified topic and forwards the message to the HTTP server.

The architecture is composed by the following three elements: message translator, protocol plugins, and graphical user interface. Message translator is the entity responsible for receiving, modifying, and forwarding the message to the desired protocol. Protocol plugins are the implementation of specific application protocols, communications between the different protocols must go through the message translator (only MQTT and HTTP are currently supported). The graphical user interface allows users to configure aspects related to the conversion and forwarding of messages during runtime.

2.6 A Lightweight Multi-tier S-MQTT Framework to Secure Communication between low-end IoT Nodes

A secured MQTT protocol in detailed analysis of data and devices security issues and present an enhanced security model with a view to improving the security issues. This secure version of MQTT protocol can modify and enhance existing MQTT protocol based on the Key/Cipher text Policy Attribute Based Encryption(KP/CP-ABE) using lightweight Elliptic Curve crypto system plus multi-tier authentication system to prevent data theft.

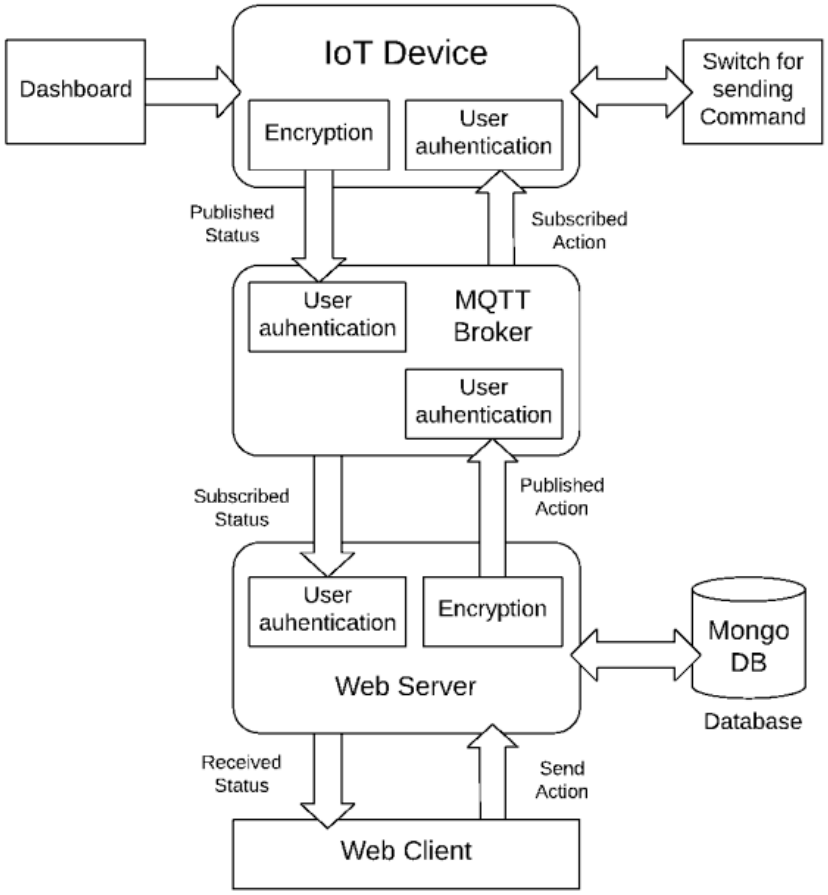


Figure 6 System Block Diagram

By using KP/CP-ABE with lightweight elliptic curve cryptography (ECC) which can secure data in multiple layers while maintaining the lightweight communication. Attribute based encryption is basically a type of public key crypto-system in which the secret sharing key and ciphertext are followed upon some particular attributes. With the usage of lightweight ECC in KP-ABE (key-policy based ABE) or CP-ABE (ciphertext-policy based ABE), a secure communication between low end device can be achieved. The objective of this secure version of MQTT are:

- Can analyze the use of cryptographic approaches in different communication protocol
- To create a secure version of lightweight MQTT protocol for wireless sensor networks
- Develop a multi-tier authentication system to ensure data privacy

2.7 MQTT protocol employing IOT based home safety system with ABE encryption

Smart home security with small percentage of usage of cloud memory and clubbed with WSNs (Wireless Sensor Networks) gives major advantages to traditionally employed systems. In networking, Wi-Fi is used and MQTT is being used for message transfer instead of HTTP protocol since MQTT has high popularity, simplicity of use and experienced open-source support. Plus, with the use of encryption methods which suitable for IoT and MQTT protocol. This MQTT protocol is used with devices which had memory, bandwidth and computational powers constraint such as new-age small size microcontrollers which can saves lot of times unlike HTTP which need to create web server then need to communicate via HTTP requests means MQTT protocol can greatly reduce load on the processor.

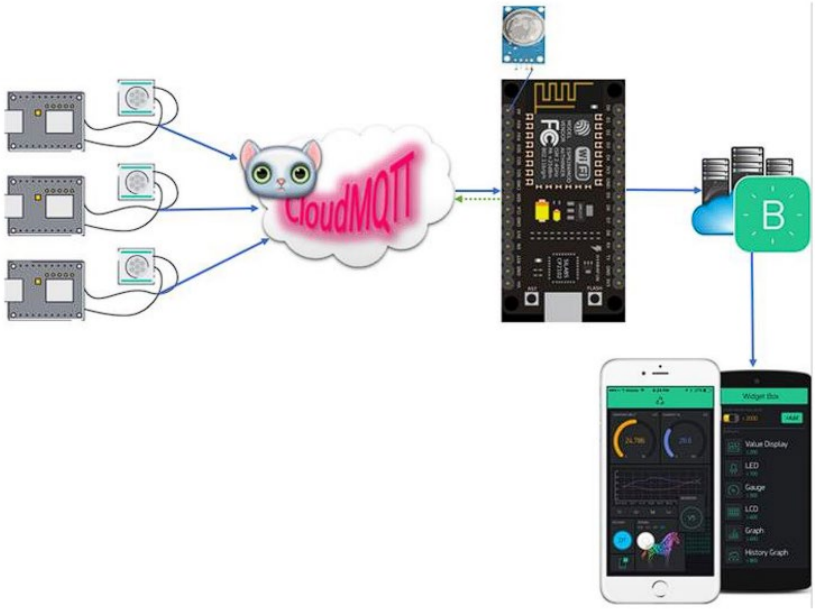


Figure 7 IoT based subscriber – publisher model

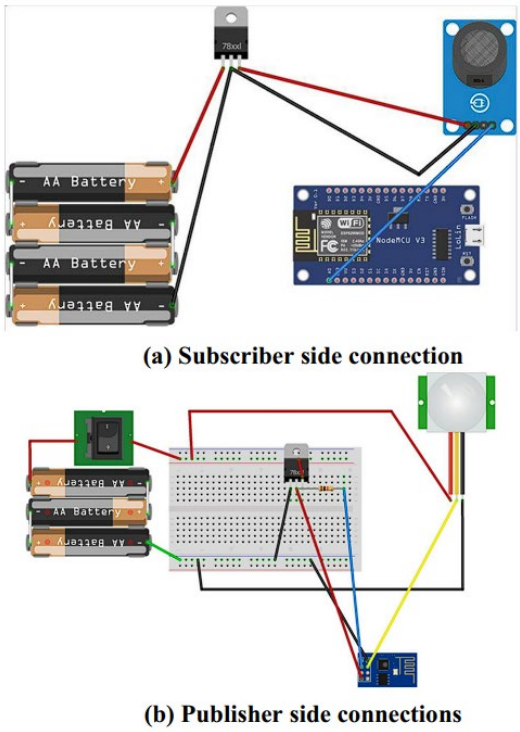


Figure 8 Physical connection of whole system with subscriber and publisher side connection

3 PIR sensors are used along with a single MQ-5 gas sensor which connected to individual ESP-01 8266 Wi-Fi modules. Microcontroller hosts by a small Wi-Fi module which can act as both a processor as well as a Wi-Fi transceiver. This sensor will collect information and with the help of Wi-Fi module connected, it will upload the collected data to the cloud MQTT broker to be access, parse and send the data to its appropriate subscriber. The MQTT broker here also functions as the PKG. Only one subscriber exists which is the NodeMCU ESP-8266 12E module which has been subscribed to all topics. MQTT is a publish-subscribe type communication protocol and the individual ESP-01 8266 which is placed at different locations publish at unique topic regarding sensor status. The actual publisher system is subscribed to all the topics which the whole network is publishing. This happened because Blynk app can be configured to only one board to have access to all smaller Wi-Fi modules. NodeMCU board will receive all the encrypted messages the modules publish in accordance with a unique topic. This decrypted data is sent to Blynk app and if any suspicious activity is detected, the app will notify user for further actions taken.

2.8 LAKA: Lightweight Authentication and Key Agreement Protocol for Internet of Things Based Wireless Body Area Network

The wireless body area network (WBAN) concept has managed to conquer the medical field by allowing hospitals, patients, and doctors to exchange critical health data for efficient medical services. Wireless Body Area Network (WBAN) is a unique type of sensor network that connects patients with medical service providers to remotely exchange critical health data. WBAN is an important wearable and implant network that detects various vital data from various wireless sensors (deployed in/over the body) for health diagnosis, monitoring, and controlling actuators. It has several advantages, including location independence, no impact on patients' mobility, early disease detection and prevention, remote patient assistance, and so on. As a result, it is extremely useful and effective for continuous monitoring, allowing for accurate diagnostics and real-time feedback to medical professionals.

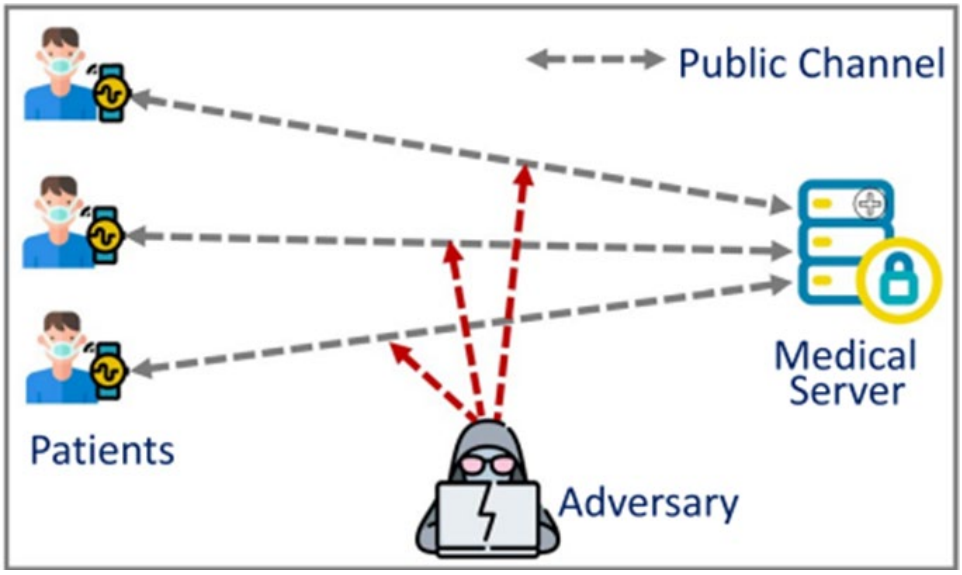


Figure 9 Shows the exchange system between patients and the medical server

Because data is sent over a public channel, an adversary can intercept transferred messages and use them to carry out various malicious activities such as data modification, data delay, information impersonation, and data exchange interruption. Each of these issues has the potential to disrupt real-time data transmission in the medical field. Furthermore, as medical data is transferred from a patient to a medical server over a public network, it is critical to maintain user anonymity during the data communication phase in order to protect user privacy. Furthermore, wireless sensors have limited computing power to perform various data exchange operations, necessitating the development of a low-cost WBAN protocol.

2.9 A Lightweight IoT Security Protocol

IoT is a technology that connects between smart physical and virtual entities also it offers cutting-edge services. The entities generally are constrained devices that are limited by their computing power, storage and capacity, also the energy they use. A Wireless Sensor Network (WSN) is a network made up of devices controlled by a CPAN (Personal Area Network Coordinator). The network's purpose is to gather and process the data according to their environment. For the purpose of protecting the Wireless Sensor Network (WSN), mutual authentication between devices should be completed before accepting any new device into the network. The transmitted data or information must be authenticated and encrypted to ensure secure communication.

The Personal Area Network Coordinator (CPAN), an unconstrained device, manages the constrained devices (sensors or actuators) that make up the Wireless Sensor Networks (WSN) architecture. In order to join the WSN network, a new device must first facilitate mutual authentication. Then, in order to secure the transmitted data, a symmetric secured channel should be established between the communicating entities. The MAC sub-layer implemented the mutual authentication mechanism, and the application layer performs the authenticated data encryption.

- Chosen Algorithms

The asynchronous OTP (One Time Password) algorithm will be used for the authentication mechanism. The OTP is self-explanatory, it is a password that can be used only once. The asynchronous OTP is based on a random challenge and pre-shared key. This can ensure that the authentication will be secured against cryptanalysis attacks and replay attacks. The AES has been implemented for fast and robust authenticated encryption of data. Because of the authenticated encryption, it can protect confidentiality and the integrity of the exchange data within the same time. The OCARI network (Optimization of Communication for Ad hoc Reliable Industrial networks) was the target where this protocol was deployed. It is an illustration of an IoT application in an industrial environment. The physical layer of IEEE 802.15.4, on which OCARI is based on, enables reliable signal transmission and resistance to radio interferences in extreme environments (power plants, factories, etc.). Although OCARI is the WSN for which this solution is suggested and implemented, it can be used with any other WSN.

- Protocol Design

A method called “personalization” has been created for key management. The principle of “personalization” method is detailed in the figure below.

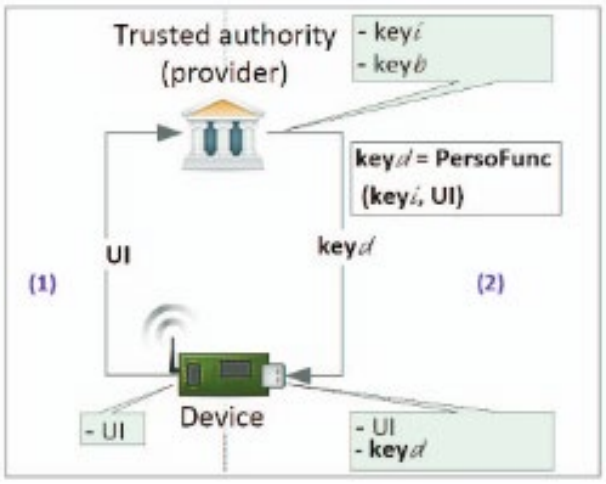


Figure 10 The personalization of devices

The devices provider generates a "kit" of secret keys that contains the initial key key_i and derived one $key_d(s)$. The kit will be installed in the CPAN and the relevant devices in out-of-band mode. The derived keys are computed from the unique identifier (UI) of every device and key_i by using the "PersoFunc()" function. This function, which is irreversible, creates a robust key and secures the key_i against deductive attacks.

$$persoFunc(key_i, UI) = HMAC(key_i, UI)$$

Figure 11 "PersoFunc()"

The device can be connected to the OCARI network once the key_d has been generated and fixed into the device.

This personalization aims to prevent device B connected to the same OCARI network from listening in on communication between device A and the CPAN. Another benefit is that even if an attacker were to obtain a personalized key for one device, it would not have an impact on the security of the other devices connected to the same OCARI network.

- Authenticated Encryption

Firstly, data that wants to be transmitted by the entity must be authenticated and encrypted by using the generated key_u in unicast mode and key_b in broadcast mode. A plaintext $P_1..P_n$, an additional authenticated data A (A can be any random data that was added to strengthen the encryption algorithm) and an initialization vector (IV) as inputs was required in this operation. The key_u and a counter value will decide the generation of the IV. The latter is used in order to avoid cryptanalysis attacks. The ciphertext $C_1..C_n$ and an authenticated tag T are obtained as the result.

The authenticated encryption operation was defined by the equations figure below.

$$\begin{cases} H = E(K, 0^{128}) \\ \text{We use } len(IV) \text{ of 96 bits} \\ \quad => Y_0 = IV||0^{31}1 \\ Y_i = Y_{i-1} + 1, for\, i = 1..n \\ C_i = P_i \oplus E(K, Y_i), for\, i = 1, ..n \\ T = MSB_t(GHASH(H, A, C, len(A), len(C)) \\ \quad \oplus E(K, Y_0)) \end{cases}$$

Figure 12 Authenticated Encryption operation equations

E – Encryption operation

K – Secret key (K_u or K_b)

0^{128} – Block of 128 bits of 0

H – Obtained by encrypting a zero block using *K*

Y – Counter starting from Y_0 which is the concatenation (||) of *IV* with 31 zeros and one (bits)

MSB_t – Most significant bit

len(A) - The length of *A*

GHASH() - Hash function of the GCM mode of operation

The *ciphertext* and the tag *T* are to be commonly concatenated at the end of the authenticated encryption operation. Once the packet is received, the latter will be used to check the message's integrity.

- Authenticated Decryption

There are four input parameters for the authenticated decryption operation, which are the ciphertext *C*, the received tag *T*, the *IV* and the authenticated additional data *A*. For the output, the plaintext *P* and a Tag *T'* are obtained to checking the data integrity. When compared to the encryption operation, the order of the hash step and decrypt step are reversed.

The authenticated decryption operation was defined by the equations figure below.

$$
\begin{cases}
H = E(K, 0^{128}) \\
\text{We use } len(IV) \text{ of 96 bits} \\
\quad => Y_0 = IV||0^{31}1 \\
T' = MSB_t(GHASH(H, A, C, len(A), \\
\quad len(C)) \oplus E(K, Y_0)) \\
Y_i = Y_{i-1} + 1, for\, i = 1..n \\
P_i = C_i \oplus E(K, Y_i), for\, i = 1, ..n
\end{cases}
$$

Figure 13 Authenticated Decryption operation equations

The received *T* and *T'* will be compared at the end of the authenticated decryption operation. If the result is a match, the decryption operation is a success otherwise it fails.

CHAPTER 3: COMPARISON AMONG TECHNOLOGIES

3.1 Comparison between MQTT and Http as an IoT protocol

In an IoT environment, Http works on TCP/IP which provides a reliable communication. Http sends many small packets to the server to get connected which can possibly cause a large overhead. Http calls are stateless which lead to doing authentication every time it connected to IP or URL to do API calls because after the response the device will close the connection. The Http request uses complex header format of TCP with 9 packets which helps human readability but are not required in most cases of IoT as it is an unnecessary waste of resources. Due to these reasons, Http protocol commonly used where data is triggered by the client like weather reporting and pollution status on a timely basis.

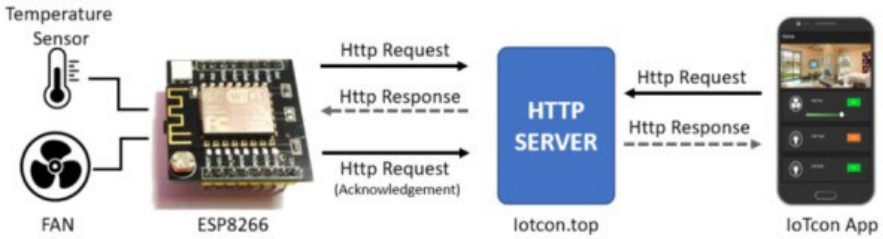


Figure 14 Example of Http protocol in IoT environment

In contrast, MQTT works on the principle of Publish / Subscribe which logically attaches a client (subscribing) to topic of interest so that it can exchange messages with this topic. It becomes beneficial to use this protocol in the scenario where we want to exchange small messages which require less bandwidth. MQTT also has a very small message header and packet size which guarantees a fast delivery even on low bandwidth network. An example of MQTT protocol use case is a smart factory where installed temperature sensors connect to the MQTT broker and publish the data within sensor topics. With these specifications, MQTT is a good choice of protocol for many IoT environments.

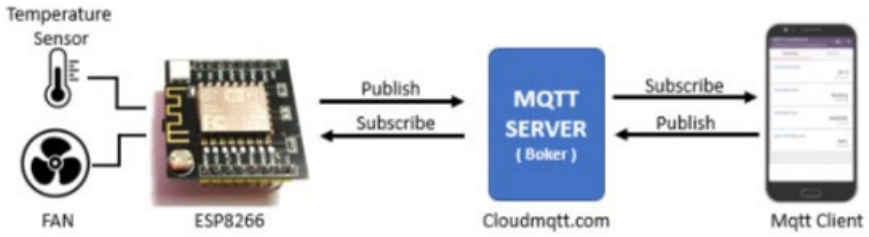


Figure 15 Example of MQTT protocol in IoT environment

CHAPTER 4: OPEN ISSUE AND RECOMMENDATION

4.1 A Lightweight Authentication Protocol for Internet of Things

With an emphasis on cryptographic protocols and the IoT, the security procedures for the current RFID system can be improved. The current lightweight cryptography protocol can be used to strengthen the original RFID protocols' shortcomings or weaknesses. The hardware implementation of the lightweight cryptography protocol is demonstrated in this paper. In addition, the proposed protocol can be used to establish the mutual authentication procedure in a typical RFID system for IOT applications.

4.2 Lightweight Authentication and Key Agreement Protocol for Internet of Things Based Wireless Body Area Network

Existing schemes are prone to diverse safety threats together with impersonation, modification, replay, clever card loss, and different applicable threats. Furthermore, in an effort to shield opposition from an adversary, person variables ought to be stored in a Tamper Resistance Memory (TRM), which increases the preliminary value of setup in addition to the value of blanketed garage memory. Furthermore, current WBAN authentication mechanisms require a whole lot of computational energy due to the fact they use high-priced cryptographic operations. When facts are transmitted from the affected person to the server, the transmission postponement increases.

To resolve the issue, suggest an authentication and key settlement mechanism for data trade in BAN using a low-price function. Furthermore, implement a security evaluation to ensure the robustness of the towards crucial safety attacks. Besides, conduct an overall performance evaluation to decide the price of execution, communication costs, and power intake costs, in addition, to examining the overall performance consequences of the suggested mechanism.

CHAPTER 5: CONCLUSION

In a nutshell, LWP in IoT application is very important in order for developer, technologist or industrial people either to create new device or to upgrade their products technology system to be able to operate efficiently accordance with advanced technologies nowadays. With LWP, the communication among devices can be established, so that the devices can exchange data, sending command, transmit useful information either among the devices or device to end user. MQTT protocol is a common example of LWP in IoT application. MQTT protocol used for exchanging data between devices in the Internet of Things. Another example is Constrained Application Protocol (CoAP). CoAP is an IETF standard RFC 7252 created by the Constrained Environments Working Group (CoRE) for use in low power devices of class 0, 1, or 2, as specified in RFC 7228. All these protocols are design correspondingly for their own use or functionality.

Lots of devices use multiple lightweight protocol that can make the devices be able to work well in their own IoT system. Some of the LWP that have been used are LoWPAN, ZigBee, Bluetooth, RFID, NFC, WiFi, Z-Waze, Cellular. Each IoT protocols have their work functionality, standard, characteristic as well as security. Based on each criterion of LWP, developer must determine the right protocol which are the best for their loT system.

As lightweight wireless protocol for IoT applications have attracted the industry's interest, IoT systems have a greater number of inherent flaws. The more usage in IoT application, security demands will increase. From these, LWP issues and concern occur. LWP issues need to be emphasized in order minimize the flaws happen. Some of the issues are talk about security and privacy issues, data transmission latency, multiple vendors, network bandwidth, compatibility and longevity, data complexity as well as data volume. From these issues, the are some recommendations such as smart city, retail and logistic, healthcare as well as security. Lastly, the growth of lightweight wireless protocol in Internet of Things (loT) applications is very important in technologies industries and also in our daily live.

REFERENCES

- Al-Fuqaha, M. G. (2015). Internet of Things: A survey on enabling technologies, protocols, and. *IEEE Commun. Surveys Tuts.*
- Aman, M. N. (2020). A lightweight protocol for secure data provenance in the Internet of Things using wireless fingerprints. *IEEE Systems Journal*.
- da Cruz, M. A. (2018). A proposal for bridging the message queuing telemetry transport protocol to HTTP on IoT solutions. *2018 3rd International Conference on Smart and Sustainable Technologies (SpliTech)*.
- Gaikwad, K. (2021). IoT based Water Management System using MQTT. *2021 5th International Conference on Trends in Electronics and Informatics (ICOEI)*.
- Gupta, V. K. (2021). MQTT protocol employing IOT based home safety system with ABE encryption. *Multimedia Tools and Applications*.
- Hammi, M. T. (2017). A Lightweight IoT Security Protocol. *1st cyber security in networking conference (CSNet)*.
- Lee, J. Y. (2014). A lightweight authentication protocol for internet of things. *2014 International Symposium on Next-Generation Electronics*.
- Rahman, A. R. (2018). A lightweight multi-tier S-MQTT framework to secure communication between low-end IoT nodes. *2018 5th International Conference on Networking, Systems and Security (NSysS)*.
- Raikar, M. M. (2018). Blend of Cloud and Internet of Things (IoT) in agriculture sector using lightweight protocol. *2018 international conference on advances in computing, communications and informatics (ICACCI)*.
- Sanjuan, E. B. (2020). Message Queuing Telemetry Transport (MQTT) Security: A Cryptographic Smart Card Approach. *IEEE Access 8*.
- Soni, M. &. (2021). LAKA: lightweight authentication and key agreement protocol for internet of things based wireless body area network. *Wireless Personal Communications*.
- V. Hassija, V. C. (2019). A survey on IoT security: Application areas, security threats, and solution.
- Wukkadada, B. W. (2018). Comparison with HTTP and MQTT in Internet of Things (IoT). *2018 International Conference on Inventive Research in Computing Applications (ICIRCA)*.

g4 assignment

by Ahmad Sha Herizam Bin Tahir

Submission date: 24-Nov-2022 09:53PM (UTC+0800)
Submission ID: 1962597628
File name: v1_Group_4_Assignment.docx (1.55M)
Word count: 4932
Character count: 27095

BITS 3533 Wireless Network and Mobile Computing

Sem 1 2022/2023

Assignment 1

Lightweight Wireless Protocol for IoT Application

Group 4 Members:

Muhammad Izham Bin Norhamadi	B032020039
Ahmad Sha Herizam Bin Tahir	B032020009
Affendy Elyas bin Azhari Sharidan	B032020024
Muhammad Rifqi Bin Ramlan	B032020028
Muhammad Ikmal Bin Mazlan	B032020002
Muhammad Khalif Asyara Bin Noor Shansuddin	B032110015

Lecturer:

TS. DR. NORHARYATI BINTI HARUM

TEAM MEMBERS AND ROLES

Group Member	In charge of
Izham	- Chapter 1: Introduction - 2.1 MQTT with Cryptographic Smart Card - 2.5 A Proposal for Bridging the MQTT Protocol to HTTP on IoT Solutions - 3.1 Comparison between MQTT and Http as an IoT protocol
Ahmad	- 2.2 IoT based Water Management System using MQTT protocol - 2.3 Blend of Cloud and Internet of Things (IoT) in agriculture sector using lightweight protocol - 2.6 A Lightweight Multi-tier S-MQTT Framework to Secure Communication between low-end IoT Nodes - 2.7 MQTT protocol employing IOT based home safety system with ABE encryption
Affendy	- A Lightweight Authentication Protocol for Internet of Things
Rifqi	- A Lightweight IoT Security Protocol
Ikmal	- 2.8 WBAN - 5.0 Conclusion
Khalif	- LAKA: Lightweight Authentication and Key Agreement Protocol for Internet of Things Based Wireless Body Area Network - 2.8 WBAN - 4.2 WBAN open issue and recommendation

TABLE OF CONTENT

CHAPTER 1: INTRODUCTION

The Internet of Things (IoT) is an ecosystem that provides the possibility of communications on internet to countless devices of various types such as sensors, devices, gateways, servers, appliances, user applications and many more. It is expected that by the end of 2022 there will be 20.4 billion of IoT devices connected to internet (V. Hassija, 2019). End Application requirements of range, data, security, power, and battery life dictate the network protocol of choice. One of the most appropriate communications protocols for the IoT is the Message Queuing Telemetry Transport (MQTT) protocol, due to its capacity for easy implementation on lightweight, cheap, low-power, and low memory devices (A. Al-Fuqaha, 2015) which allows communication between nodes in both reliable and unreliable networks.

MQTT protocol was designed by IBM and was standardized by OASIS (Open Architecture System) in 2013 and it was approved as ISO standard called ISO/IEC 20922 from 2016. MQTT follows a publish/subscribe architecture, meaning there are nodes (Brokers) that make the information available while clients can read the available information after subscribing by accessing corresponding URL. MQTT has three types of participants: the Broker, the Publisher, and the Subscriber. The Broker is the center of a star topology in MQTT protocol and oversees the exchange of messages between other participants. Every other participant can only connect with it as it also authenticates them in the network. The Publisher are the elements that send data to the Broker so that it sends this data to one or more Subscribers that require it. The Subscribers are the elements that receive data to the Broker which are sent by the Publishers.

When Kevin Ashton was creating the foundation for the Internet of Things (IoT) at MIT's AutoID lab in 1999, he invented the term "IoT." The internet of things is quickly evolving and becoming widely used because to improvements in wireless networking technology and more standardisation of communications protocols. All everyday items could talk with one another and be controlled by computers if they were given unique identifiers and wireless connectivity. As the Internet became more commercialised, security worries grew to include issues with financial transactions, individual privacy, and the risk of cybercrime. IoT security and safety go hand in hand. The IoT has significant hurdles in terms of security and privacy.

CHAPTER 2: TECHNICAL DESCRIPTION / DESIGN

2.1 Message Queuing Telemetry Transport (MQTT) with Cryptographic Smart Card

The implementation of Cryptographic Smart Card provides hardware secure, trustworthy, well tested and with low economic cost in the IoT devices to execute all necessary cryptographic functions, and a public key repository accessible for the broker (Sanjuan, 2020). With these new elements a new method for mutual authentication was presented in the MQTT protocol. An encryption schema was defined for encoding the data exchange between the clients and the broker, in both directions without including modifications in the specification of the protocol messages.

Figure 1 Publish/subscribe model of MQTT protocol with security schema based on cryptographic smart card

2.1.1 General Scheme of Security

To achieve the objectives in the article, there are three new elements in the system.

A. PUBLISHER/SUBSCRIBER CRYPTOGRAPHIC SMART CARD

The publishers and the subscribers MQTT must have a Cryptographic Smart Card that must be communicated with a microcontroller that manages the communication with the broker.

B. BROKER CRYPTOGRAPHIC SYSTEM

The broker must have a cryptographic system, either a Cryptographic Smart Card or an HSM (Hardware Security Manager) system and must complete several cryptographic functions for all messages sent between publishers and subscribers. If the number of clients and the frequency of messages produces messages overlaps, the system should execute several cryptographic operations in parallel with high speed.

C. PUBLIC KEY REPOSITORY

The system must have a public key repository accessible, through a secure protocol, by the broker. The implementer of this system can select the asymmetric cryptography algorithm (RSA_NOPAD, RSA_PKCS1, ECC, etc.), that he prefers for authentication proposes and the block cipher algorithm (AES, DES, TEA, NOEKEON, etc.) for payload encryption.

2.2 IoT based Water Management System using MQTT protocol

Nowadays, increasing environmental pollution, world population, water scarcity, etc. are some major issues faced by the society. We need solutions, which would be implemented from a individual person to large industries (Gaikwad, 2021). The ultrasonic sensor is used to measure the level of tank, the flow sensor is used to measure the flow rate of water and total consumed volume. This system is used to monitor usage of water, avoid overflow and saves water. All the information data such as level, flow rate and volume data will be transmitted by Node MCU to mobile app using the Message Queuing Telemetry Transport (MQTT) protocol.

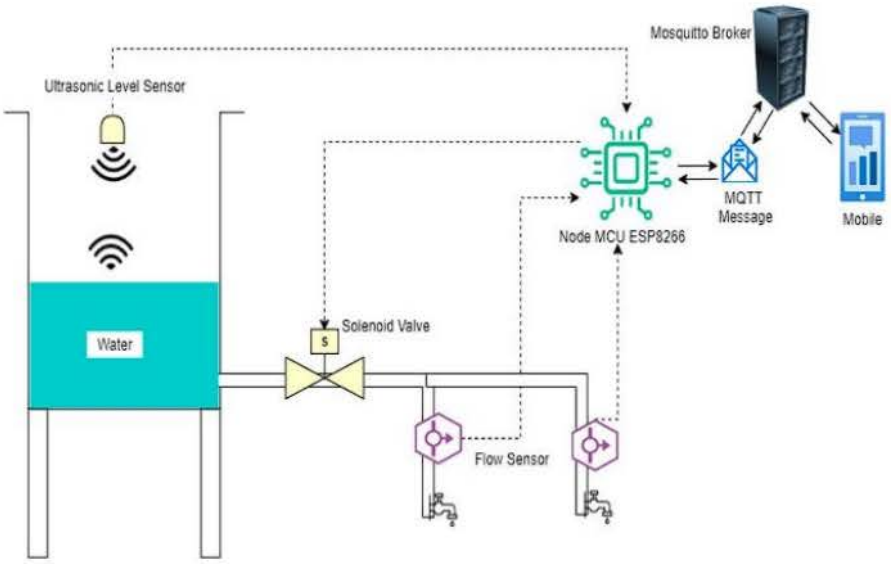


Figure 2 Water Management System using MQTT protocol

2.2.1 Technical Description

Firstly, Node MCU connected to internet through WiFi to calculates the volume, flow rate and level. After that, it sends the message to mosquitto broker under topic of 'esp/flow', 'esp/flow' and 'esp/level' respectively. Simultaneously it subscribe to topic 'esp/valve' to receive message to control valves. In any case of Node MCU failed to connect to mosquitto broker it will return code.

On mobile applications side, it will connect to MQTT mosquitto broker then subscribe to the same topic in Node MCU. Automatically, this wll make the app receive message from broker sent by Node MCU. All the messages will be stores in SQLite database and can be displayed in a graph.

2.3 Blend of Cloud and Internet of Things (IoT) in agriculture sector using lightweight protocol

A smart irrigation system is developed using the lightweight protocol, MQTT (Message Queue Telemetry Transport) since this protocol have 22% more energy efficient plus 15% faster compared to others protocol. All the information such as temperature and soil moisture data are collected and managed by Amazon Cloud. The data analysis is performed using the Weka (Waikato Environment for Knowledge Analysis) tool. With the invent of IoT and Cloud Computing, the agriculture sector has opened up a new door for "Precision agriculture". Hence, the CloudIoT based irrigation system is designed and developed to conserve more water. Smart irrigation is capable of supplying the water to the entire field uniformly, schedule and the water supply remotely so that each plant has the adequate amount of water it needs (Raikar, 2018).

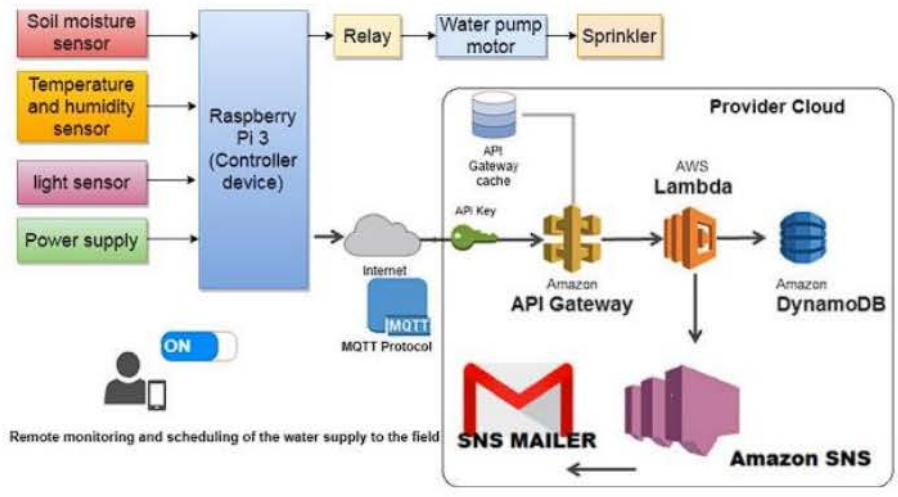


Figure 3 CloudIoT smart irrigation system

The CloudIoT architecture layers are applied to the case study smart irrigation system in the agriculture sector and divided in some parts:

- User Layer
  The usage of mobile app to turn on/off water motor remotely by user to increase water efficiency. User also can schedule the watering plants activity based on the soil moisture in the field.

- Proximity network
  IoT device: The temperature, moisture, and light intensity data are sent to the IoT gateway device
  IoT gateway device: Using the Raspberry Pi 3 as gateway device and connect it to AWS provider cloud

- Public Network
  IoT connectivity: MQTT is used for the communication between the client nodes and the service provider. MQTT protocol uses the 'publish-subscribe' type of communication model.

- Provider cloud
  Device management: MQTT server being executed using AWS IoT.
  API management: The access control keys are obtained for reading and writing data and also to implement AWS alarms and cloud watch services.
  Device data store: All sensor values is stored in the database for further referral.
  Data analysis: The Weka (Waikato Environment for Knowledge Analysis) tool used to perform data analysis such as classification and clustering,

2.4 A Lightweight Authentication Protocol for Internet of Things

A communication scenario used to creating the IOT using Radio Frequency Identification (RFID) tags via the Internet is shown on Figure 4

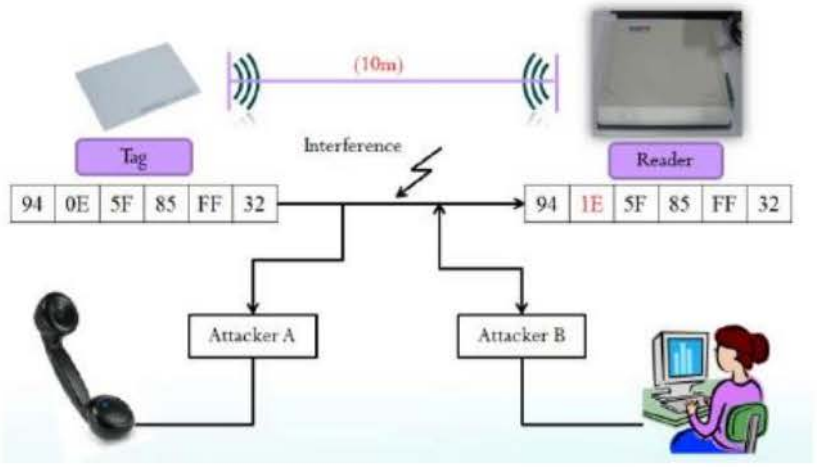


Figure 4 A communication scenario used to creating the IOT using RFID tag

The RFID readers are connected to the Internet. The tagged items are mobile and are expected to move through different reader fields and "connect" to the readers via their standard RFID communication protocol. The RFID reader identifies the tag using an appropriate authentication protocol. A real hurdle to designing secure RFID is the absence of cryptography in the technology. The Class-1 Gen-2 type of EPC tags have absolutely no specific anti-counterfeiting characteristics. In theory, a hacker could easily copy the EPC from a target tag and programmed it into a fake tag or mimic the target tag on a different kind of wireless device.

2.5 A Proposal for Bridging the MQTT Protocol to HTTP on IoT Solutions

Software that handles and stores IoT devices data is known as IoT middleware. The devices transfer data to middleware through an application protocol, which can be different from those supported by middleware. To overlap this problem, this paper proposes an application layer gateway for IoT protocols that can "translate" MQTT messages into HTTP reducing the packet size sent by an IoT device and it is fully configurable through a graphical user interface in runtime (da Cruz, 2018). The software responsible for such conversion is called application layer bridge (ALB), application layer gateway (ALG), or simply bridge.

Figure 5 Illustration of a bridge operation where a gateway messages through a protocol A translates them to a protocol B

2.5.3 Bridging from MQTT to HTTP

The application layer gateway presented in this paper translates MQTT requests into HTTP. Its primary goal is to allow constrained devices to send fewer data when communicating with IoT middleware. The solution can be deployed on any device with Java installed which are widely used programming language. To use the solution, users must specify the MQTT broker details (IP, port, topic) and the middleware server details (IP, port, path). The software then subscribes to the specified topic and forwards the message to the HTTP server.

The architecture is composed by the following three elements: message translator, protocol plugins, and graphical user interface. Message translator is the entity responsible for receiving, modifying, and forwarding the message to the desired protocol. Protocol plugins are the implementation of specific application protocols, communications between the different protocols must go through the message translator (only MQTT and HTTP are currently supported). The graphical user interface allows users to configure aspects related to the conversion and forwarding of messages during runtime.

2.6 A Lightweight Multi-tier S-MQTT Framework to Secure Communication between low-end IoT Nodes

A secured MQTT protocol in detailed analysis of data and devices security issues and present an enhanced security model with a view to improving the security issues. This secure version of MQTT protocol can modify and enhance existing MQTT protocol based on the Key/Cipher text Policy Attribute Based Encryption(KP/CP-ABE) using lightweight Elliptic Curve crypto system plus multi-tier authentication system to prevent data theft.

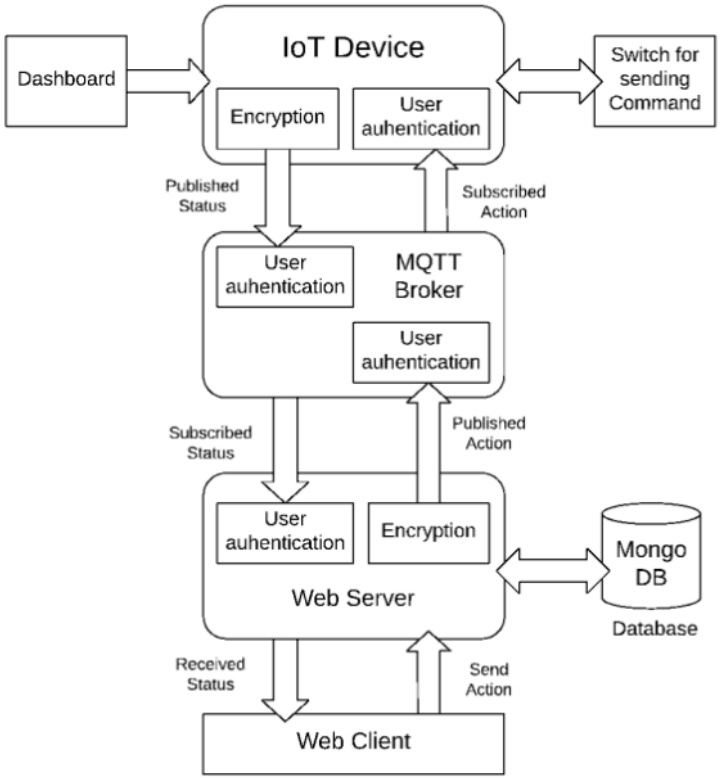


Figure 5 System Block Diagram

By using KP/CP-ABE with lightweight elliptic curve cryptography (ECC) which can secure data in multiple layers while maintaining the lightweight communication. Attribute based encryption is basically a type of public key crypto-system in which the secret sharing key and ciphertext are followed upon some particular attributes. With the usage of lightweight ECC in KP-ABE (key-policy based ABE) or CP-ABE (ciphertext-policy based ABE), a secure communication between low end device can be achieved. The objective of this secure version of MQTT are:

- Can analyse the use of cryptographic approaches in different communication protocol
- To create a secure version of lightweight MQTT protocol for wireless sensor networks
- Develop a multi-tier authentication system to ensure data privacy

2.7 MQTT protocol employing IOT based home safety system with ABE encryption

Smart home security with small percentage of usage of cloud memory and clubbed with WSNs (Wireless Sensor Networks) gives major advantages to traditionally employed systems. In networking, Wi-Fi is used and MQTT is being used for message transfer instead of HTTP protocol since MQTT has high popularity, simplicity of use and experienced open-source support. Plus, with the use of encryption methods which suitable for IoT and MQTT protocol. This MQTT protocol is used with devices which had memory, bandwidth and computational powers constraint such as new-age small size microcontrollers which can saves lot of times unlike HTTP which need create web server then need to communicate via HTTP requests means MQTT protocol can greatly reduce load on the processor.

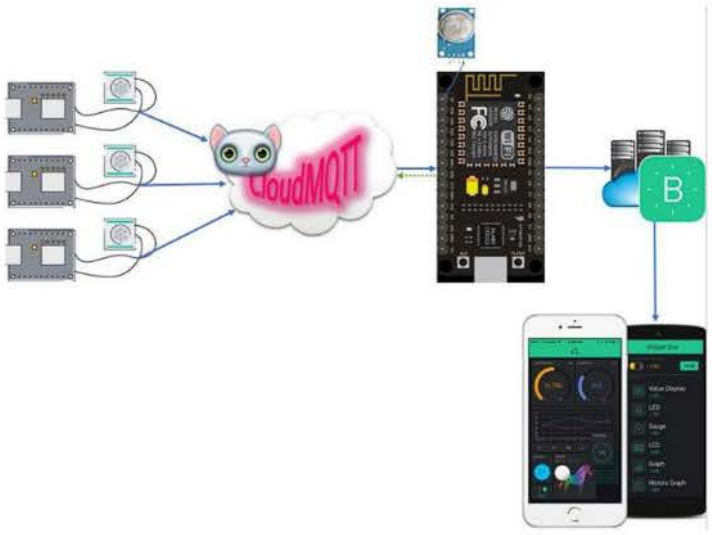
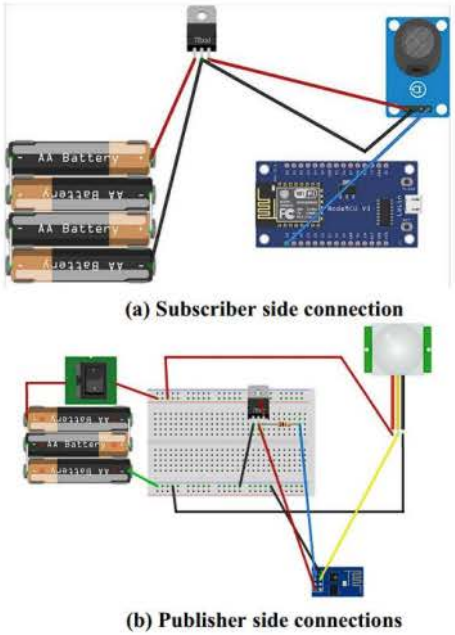


Figure 7 IoT based subscriber – publisher model



(a) Subscriber side connection

(b) Publisher side connections

Figure 8 Physical connection of whole system with subscriber and publisher side connection

3 PIR sensors are used along with a single MQ-5 gas sensor which connected to individual ESP-01 8266 Wi-Fi modules. Microcontroller hosts by a small Wi-Fi module which can act as both a processor as well as a Wi-Fi transceiver. This sensor will collect information and with the help of Wi-Fi module connected, it will upload the collected data to the cloud MQTT broker to be access, parse and send the data to its appropriate subscriber. The MQTT broker here also functions as the PKG. Only one subscriber exists which is the NodeMCU ESP-8266 12E module which has been subscribed to all topics. MQTT is a publish-subscribe type communication protocol and the individual ESP-01 8266 which is placed at different locations publish at unique topic regarding sensor status. The actual publisher system is subscribed to all the topics which the whole network is publishing. This happened because Blynk app can be configured to only one board to have access to all smaller Wi-Fi modules. NodeMCU board will receive all the encrypted messages the modules publish in accordance with a unique topic. This decrypted data is sent to Blynk app and if any suspicious activity is detected, the app will notify user for further actions taken.

2.8 LAKA: Lightweight Authentication and Key Agreement Protocol for Internet of Things Based Wireless Body Area Network

The wireless body area network (WBAN) concept has managed to conquer the medical field by allowing hospitals, patients, and doctors to exchange critical health data for efficient medical services. Wireless Body Area Network (WBAN) is a unique type of sensor network that connects patients with medical service providers to remotely exchange critical health data. WBAN is an important wearable and implant network that detects various vital data from various wireless sensors (deployed in/over the body) for health diagnosis, monitoring, and controlling actuators. It has several advantages, including location independence, no impact on patients' mobility, early disease detection and prevention, remote patient assistance, and so on. As a result, it is extremely useful and effective for continuous monitoring, allowing for accurate diagnostics and real-time feedback to medical professionals.

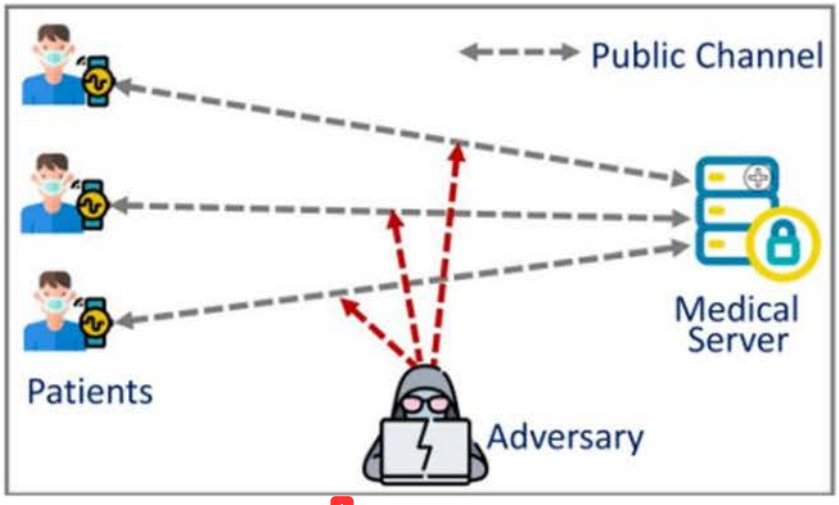


Figure 9 show the exchange system between patients and the medical server.

Because data is sent over a public channel, an adversary can intercept transferred messages and use them to carry out various malicious activities such as data modification, data delay, information impersonation, and data exchange interruption. Each of these issues has the potential to disrupt real-time data transmission in the medical field. Furthermore, as medical data is transferred from a patient to a medical server over a public network, it is critical to maintain user anonymity during the data communication phase in order to protect user privacy. Furthermore, wireless sensors have limited computing power to perform various data exchange operations, necessitating the development of a low-cost WBAN protocol.

2.9 A Lightweight IoT Security Protocol

IoT is a technology that connects between smart physical and virtual entities also it offers cutting-edge services. The entities generally are constrained devices that are limited by their computing power, storage and capacity, also the energy they use. A Wireless Sensor Network (WSN) is a network made up of devices controlled by a CPAN (Personal Area Network Coordinator). The network's purpose is to gather and process the data according to their environment. For the purpose of protecting the Wireless Sensor Network (WSN), mutual authentication between devices should be completed before accepting any new device into the network. The transmitted data or information must be authenticated and encrypted to ensure secure communication.

The Personal Area Network Coordinator (CPAN), an unconstrained device, manages the constrained devices (sensors or actuators) that make up the Wireless Sensor Networks (WSN) architecture. In order to join the WSN network, a new device must first facilitate mutual authentication. Then, in order to secure the transmitted data, a symmetric secured channel should be established between the communicating entities. The MAC sub-layer implemented the mutual authentication mechanism, and the application layer performs the authenticated data encryption.

- Chosen Algorithms

The asynchronous OTP (One Time Password) algorithm will be used for the authentication mechanism. The OTP is self-explanatory, it is a password that can be used only once. The asynchronous OTP is based on a random challenge and pre-shared key. This can ensure that the authentication will be secured against cryptanalysis attacks and replay attacks. The AES has been implemented for fast and robust authenticated encryption of data. Because of the authenticated encryption, it can protect confidentiality and the integrity of the exchange data within the same time. The OCARI network (Optimization of Communication for Ad hoc Reliable Industrial networks) was the target where this protocol was deployed. It is an illustration of an IoT application in an industrial environment. The physical layer of IEEE 802.15.4, on which OCARI is based on, enables reliable signal transmission and resistance to radio interferences in extreme environments (power plants, factories, etc.). Although OCARI is the WSN for which this solution is suggested and implemented, it can be used with any other WSN.

- Protocol Design

A method called “personalization” has been created for key management. The principle of “personalization” method is detailed in the figure below.

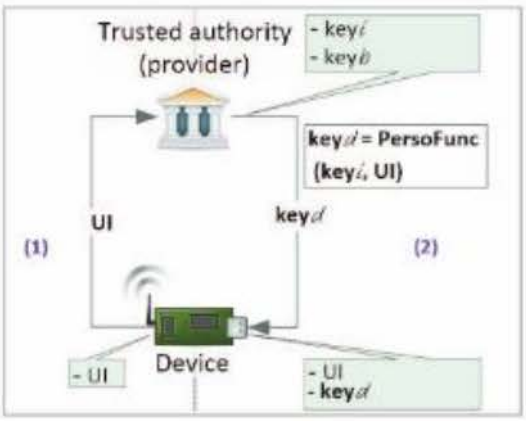


Figure 1: The personalization of devices

The devices provider generates a "kit" of secret keys that contains the initial key key_i and derived one key_d). The kit will be installed in the CPAN and the relevant devices in out-of-band mode. The derived keys are computed from the unique identifier (UI) of every device and key_i by using the "PersoFunc()" function. This function, which is irreversible, creates a robust key and secures the key_i against deductive attacks.

$$persoFunc(key_i, UI) = HMAC(key_i, UI)$$

Figure 2: "PersoFunc()"

The device can be connected to the OCARI network once the key_d has been generated and fixed into the device.

This personalization aims to prevent device B connected to the same OCARI network from listening in on communication between device A and the CPAN. Another benefit is that even if an attacker were to obtain a personalized key for one device, it would not have an impact on the security of the other devices connected to the same OCARI network.

- Authenticated Encryption

Firstly, data that wants to be transmitted by the entity must be authenticated and encrypted by using the generated key_u in unicast mode and key_b in broadcast mode. A plaintext $P_1..P_n$, an additional authenticated data A (A can be any random data that was added to strengthen the encryption algorithm) and an initialization vector (IV) as inputs was required in this operation. The key_u and a counter value will decide the generation of the IV. The latter is used in order to avoid cryptanalysis attacks. The ciphertext $C_1..C_n$ and an authenticated tag T are obtained as the result.

The authenticated encryption operation was defined by the equations figure below.

$$\begin{cases} H = E(K, 0^{128}) \\ \text{We use } len(IV) \text{ of 96 bits} \\ \quad => Y_0 = IV||0^{31}1 \\ Y_i = Y_{i-1} + 1, for\, i = 1..n \\ C_i = P_i \oplus E(K, Y_i), for\, i = 1, ..n \\ T = MSB_t(GHASH(H, A, C, len(A), len(C)) \\ \quad \oplus E(K, Y_0)) \end{cases}$$

Figure 3: Authenticated Encryption operation equations.

E – Encryption operation

K – Secret key (K_u or K_b)

0^{128} – Block of 128 bits of 0

H – Obtained by encrypting a zero block using K

Y – Counter starting from Y_0 which is the concatenation (||) of IV with 31 zeros and one (bits)

MSB_t – Most significant bit

len(A) - The length of A

GHASH() - Hash function of the GCM mode of operation

The *ciphertext* and the tag T are to be commonly concatenated at the end of the authenticated encryption operation. Once the packet is received, the latter will be used to check the message's integrity.

- Authenticated Decryption

There are four input parameters for the authenticated decryption operation, which are the ciphertext C, the received tag T, the IV and the authenticated additional data A. For the output, the plaintext P and a Tag T' are obtained to checking the data integrity. When compared to the encryption operation, the order of the hash step and decrypt step are reversed.

The authenticated decryption operation was defined by the equations figure below.

$$
\begin{cases}
H = E(K, 0^{128}) \\
\text{We use } len(IV) \text{ of 96 bits} \\
\quad => Y_0 = IV||0^{31}1 \\
T' = MSB_t(GHASH(H, A, C, len(A), \\
\quad len(C)) \oplus E(K, Y_0)) \\
Y_i = Y_{i-1} + 1, for\, i = 1..n \\
P_i = C_i \oplus E(K, Y_i), for\, i = 1, ..n
\end{cases}
$$

Figure 4: Authenticated Decryption operation equations.

The received T and T' will be compared at the end of the authenticated decryption operation. If the result is a match, the decryption operation is a success otherwise it fails.

CHAPTER 3: COMPARISON AMONG TECHNOLOGIES

3.1 Comparison between MQTT and Http as an IoT protocol

In an IoT environment, Http works on TCP/IP which provides a reliable communication. Http sends many small packets to the server to get connected which can possibly cause a huge overhead. Http calls are stateless which lead to doing authentication every time it connected to IP or URL to do API calls because after the response the device will close the connection. The Http request uses complex header format of TCP with 9 packets which helps human readability but are not required in most cases of IoT as it is an unnecessary waste of resources. Due to these reasons, Http protocol commonly used where data is triggered by the client like weather reporting and pollution status on a timely basis.

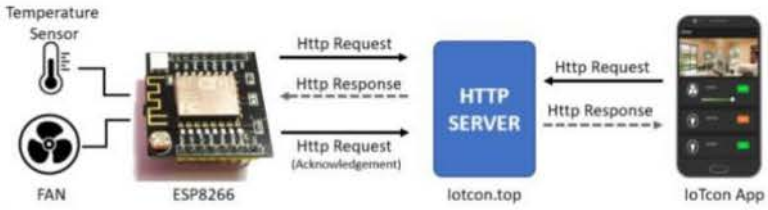


Figure 6 Example of Http protocol in IoT environment

In contrast, MQTT works on the principle of Publish / Subscribe which logically attaches a client (subscribing) to topic of interest so that it can exchange messages with this topic. It becomes beneficial to use this protocol in the scenario where we want to exchange small messages which require less bandwidth. MQTT also has a very small message header and packet size which guarantees a fast delivery even on low bandwidth network. An example of MQTT protocol use case is a smart factory where installed temperature sensors connect to the MQTT broker and publish the data within sensor topics. With these specifications, MQTT is a good choice of protocol for many IoT environments.

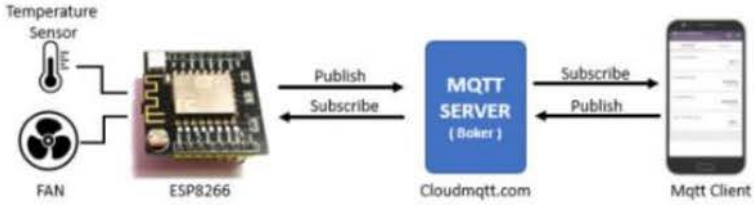


Figure 7 Example of MQTT protocol in IoT environment

CHAPTER 4: OPEN ISSUE AND RECOMMENDATION

4.1 A Lightweight Authentication Protocol for Internet of Things

With an emphasis on cryptographic protocols and the IoT, the security procedures for the current RFID system can be improved. The current lightweight cryptography protocol can be used to strengthen the original RFID protocols' shortcomings or weaknesses. The hardware implementation of the lightweight cryptography protocol is demonstrated in this paper. In addition, the proposed protocol can be used to establish the mutual authentication procedure in a typical RFID system for IOT applications.

4.2 Lightweight Authentication and Key Agreement Protocol for Internet of Things Based Wireless Body Area Network

Existing schemes are prone to diverse safety threats together with impersonation, modification, replay, clever card loss, and different applicable threats. Furthermore, in an effort to shield opposition from an adversary, person variables ought to be stored in a Tamper Resistance Memory (TRM), which increases the preliminary value of setup in addition to the value of blanketed garage memory. Furthermore, current WBAN authentication mechanisms require a whole lot of computational energy due to the fact they use high-priced cryptographic operations. When facts are transmitted from the affected person to the server, the transmission postponement increases.

To resolve the issue, suggest an authentication and key settlement mechanism for data trade in BAN using a low-price function. Furthermore, implement a security evaluation to ensure the robustness of the towards crucial safety attacks. Besides, conduct an overall performance evaluation to decide the price of execution, communication costs, and power intake costs, in addition, to examining the overall performance consequences of the suggested mechanism.

CHAPTER 5: CONCLUSION

In a nutshell, LWP in IoT application is very important in order for developer, technologist or industrial people either to create new device or to upgrade their products technology system to be able to operate efficiently accordance with advanced technologies nowadays. With LWP, the communication among devices can be established, so that the devices can exchange data, sending command, transmit useful information either among the devices or device to end user. MQTT protocol is a common example of LWP in IoT application. MQTT protocol used for exchanging data between devices in the Internet of Things. Another example is Constrained Application Protocol (CoAP). CoAP is an IETF standard RFC 7252 created by the Constrained Environments Working Group (CoRE) for use in low power devices of class 0, 1, or 2, as specified in RFC 7228. All these protocols are design correspondingly for their own use or functionality.

Lots of devices use multiple lightweight protocol that can make the devices be able to work well in their own IoT system. Some of the LWP that have been used are LoWPAN, ZigBee, Bluetooth, RFID, NFC, WiFi, Z-Waze, Cellular. Each IoT protocols have their work functionality, standard, characteristic as well as security. Based on each criterion of LWP, developer must determine the right protocol which are the best for their loT system.

As lightweight wireless protocol for IoT applications have attracted the industry's interest, IoT systems have a greater number of inherent flaws. The more usage in IoT application, security demands will increase. From these, LWP issues and concern occur. LWP issues need to be emphasized in order minimize the flaws happen. Some of the issues are talk about security and privacy issues, data transmission latency, multiple vendors, network bandwidth, compatibility and longevity, data complexity as well as data volume. From these issues, the are some recommendations such as smart city, retail and logistic, healthcare as well as security. Lastly, the growth of lightweight wireless protocol in Internet of Things (loT) applications is very important in technologies industries and also in our daily live.

REFERENCES

- Al-Fuqaha, M. G. (2015). Internet of Things: A survey on enabling technologies, protocols, and. *IEEE Commun. Surveys Tuts.*
- Aman, M. N. (2020). A lightweight protocol for secure data provenance in the Internet of Things using wireless fingerprints. *IEEE Systems Journal.*
- da Cruz, M. A. (2018). A proposal for bridging the message queuing telemetry transport protocol to HTTP on IoT solutions. *2018 3rd International Conference on Smart and Sustainable Technologies (SpliTech).*
- Gaikwad, K. (2021). IoT based Water Management System using MQTT. *2021 5th International Conference on Trends in Electronics and Informatics (ICOEI).*
- Gupta, V. K. (2021). MQTT protocol employing IOT based home safety system with ABE encryption. *Multimedia Tools and Applications.*
- Hammi, M. T. (2017). A Lightweight IoT Security Protocol. *1st cyber security in networking conference (CSNet).*
- Lee, J. Y. (2014). A lightweight authentication protocol for internet of things. *2014 International Symposium on Next-Generation Electronics.*
- Rahman, A. R. (2018). A lightweight multi-tier S-MQTT framework to secure communication between low-end IoT nodes. *2018 5th International Conference on Networking, Systems and Security (NSysS).*
- Raikar, M. M. (2018). Blend of Cloud and Internet of Things (IoT) in agriculture sector using lightweight protocol. *2018 international conference on advances in computing, communications and informatics (ICACCI).*
- Sanjuan, E. B. (2020). Message Queuing Telemetry Transport (MQTT) Security: A Cryptographic Smart Card Approach. *IEEE Access 8.*
- Soni, M. &. (2021). LAKA: lightweight authentication and key agreement protocol for internet of things based wireless body area network. *Wireless Personal Communications.*
- V. Hassija, V. C. (2019). A survey on IoT security: Application areas, security threats, and solution.
- Wukkadada, B. W. (2018). Comparison with HTTP and MQTT in Internet of Things (IoT). *2018 International Conference on Inventive Research in Computing Applications (ICIRCA).*

g4 assignment

ORIGINALITY REPORT

53%	31%	52%	%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	link.springer.com Internet Source	10%
2	Eduardo Buetas, Ismael Abad, Jose A. Cerrada, Carlos Cerrada. "Message Queuing Telemetry Transport (MQTT) Security: A Cryptographic Smart Card Approach", IEEE Access, 2020 Publication	8%
3	hal.archives-ouvertes.fr Internet Source	7%
4	Meenaxi M Raikar, Padmashree Desai, Namita Kanthi, Sachin Bawoor. "Blend of Cloud and Internet of Things (IoT) in agriculture sector using lightweight protocol", 2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI), 2018 Publication	4%
5	Kundan Gaikwad. "IoT based Water Management System using MQTT protocol",	4%

2021 5th International Conference on Trends in Electronics and Informatics (ICOEI), 2021
Publication

6	Jun-Ya Lee, Wei-Cheng Lin, Yu-Hung Huang. "A lightweight authentication protocol for Internet of Things", 2014 International Symposium on Next-Generation Electronics (ISNE), 2014 Publication	3%
7	coek.info Internet Source	3%
8	Abdur Rahman, Shanto Roy, M Shamim Kaiser, Md. Shahidul Islam. "A Lightweight Multi-tier S-MQTT Framework to Secure Communication between low-end IoT Nodes", 2018 5th International Conference on Networking, Systems and Security (NSysS), 2018 Publication	3%
9	ieeexplore.ieee.org Internet Source	3%
10	Bharati Wukkadada, Kirti Wankhede, Ramith Nambiar, Amala Nair. "Comparison with HTTP and MQTT In Internet of Things (IoT)", 2018 International Conference on Inventive Research in Computing Applications (ICIRCA), 2018 Publication	3%

11	www.allaboutcircuits.com Internet Source	1 %
12	Eduardo Buetas Sanjuan, Ismael Abad Cardiel, Jose A. Cerrada, Carlos Cerrada. "Message Queuing Telemetry Transport (MQTT) Security: A Cryptographic Smart Card Approach", IEEE Access, 2020 Publication	1 %
13	ijece.iaescore.com Internet Source	1 %
14	Mohamed Tahar Hammi, Erwan Livolant, Patrick Bellot, Ahmed Serhrouchni, Pascale Minet. "A lightweight IoT security protocol", 2017 1st Cyber Security in Networking Conference (CSNet), 2017 Publication	1 %
15	www.hindawi.com Internet Source	1 %
16	Jung Tae Kim. "Analyses of secure authentication scheme for smart home system based on internet on things", 2017 International Conference on Applied System Innovation (ICASI), 2017 Publication	<1 %
17	phdservices.org Internet Source	<1 %

18	vn.element14.com Internet Source	<1 %
19	"Connectivity Frameworks for Smart Devices", Springer Science and Business Media LLC, 2016 Publication	<1 %
20	Mauro A.A. da Cruz, Joel J.P.C. Rodrigues, Pascal Lorenz, Petar Solic, Jalal Al-Muhtadi, Victor Hugo C. Albuquerque. "A proposal for bridging application layer protocols to HTTP on IoT solutions", Future Generation Computer Systems, 2019 Publication	<1 %
21	Vatsal Gupta, Sonam Khera, Neelam Turk. "MQTT protocol employing IOT based home safety system with ABE encryption", Multimedia Tools and Applications, 2020 Publication	<1 %
22	www.coursehero.com Internet Source	<1 %
23	www.researchgate.net Internet Source	<1 %

Exclude quotes	Off	Exclude matches	< 10 words
Exclude bibliography	On		

g4 assignment

PAGE 1

PAGE 2

PAGE 3

PAGE 4

PAGE 5

PAGE 6

PAGE 7

PAGE 8

PAGE 9

PAGE 10

PAGE 11

PAGE 12

PAGE 13

PAGE 14

PAGE 15

PAGE 16

PAGE 17

PAGE 18

PAGE 19

PAGE 20

PAGE 21

PAGE 22