Diverse Malicious Attacks and security Analysis on MQTT protocol in IoT

Bhanujyothi H C
Department of Computer Science and Engineering
GITAM School of Technology, Bengaluru campus, Karnataka, India

Vidya J
Department of Computer Science and Engineering
GITAM School of Technology, Bengaluru campus, Karnataka, India

Swasthika Jain T J
Department of Computer Science and Engineering
GITAM School of Technology, Bengaluru campus, Karnataka, India

Sahana D S
Department of Computer Science and Engineering
GITAM School of Technology, Bengaluru campus, Karnataka, India

Abstract - The Internet of Things (IoT) is a model of interconnected devices, animals, objects and machines that have ability to transfer data through network without human intervention. These devices are control from anywhere and at anytime. Middleware-based IoT application protocols play an important role in facilitating two way communication and remote control of IoT devices. Compare to other IoT application protocols Message Queuing Telemetry Protocol (MQTT) is widely used protocol. It is necessary to identify possible treats in IoT environment before taking suitable countermeasures. This paper gives idea about different attacking scenarios in MQTT protocol and also demonstrate about the security requirement, issues in protocol.

Keywords- Attacks, Internet of Things (IoT), MQTT, Protocol, Security.

I. INTRODUCTION

From last few years the world has experienced rapid advancement in technology, the likes of which has had a significant impact on our daily lives. The rise of technologies - smart phones, tablets, laptops and PCs - has cause an increase in interconnectedness through time and across the spatial dimension. Contemporary technology has moved beyond fostering only connections between humans, and now facility provides both the linkage of people to things and also things to one another to achieve a common goal [2].

Internet of Things over the internet allows communication between devices over the Internet [3]. IoT plays a vital role in smart city implementation like smart home, parking, and transportation. In IoT, five of the most prominent protocols used as a communication protocol are Hypertext Transfer Protocol (HTTP), Message Queuing Telemetry Transport Protocol (MQTT), Advanced Message Queuing Protocol (AMQP), Constrained Application Protocol (CoAP) and Extensible Messaging and Presence Protocol (XMPP). While choosing protocol for communication we need to take some considerations, they are: energy efficiency, performance, resource usage, and reliability. Moreover reliability, advanced functionalities and ability to secure multicast message are highly considered. Best protocol that includes all the considerations is MQTT protocol.

In IoT information is transmitting over the internet. The growing number of incidents of cyber criminals compromising IoT devices for launching cyber attacks indicate the adverse impact of security threats in the IoT [7]. In the IoT ecosystem, users can remotely access IoT devices by making them directly accessible from the Internet or by using application message brokers or middleware technologies. Directly exposing IoT devices to the Internet for message exchange and remote control is a major security risk, as IoT devices lack extensive security mechanisms due to resource constrains [8].

Most of the IoT devices operate from behind firewalls and use middleware or message brokers for bidirectional communication and remote control [9]. In order to achieve this bidirectional communication between IoT devices (D2D) and between devices and server/cloud (D2S), several protocols have been developed. Among them, Message Queue Telemetry Transport (MQTT) has emerged as the widely adopted protocol. This is primarily due to MQTT's low overhead and power consumption. MQTT uses an Internet facing broker server to facilitate the exchange of messages between the clients which are typically IoT devices, smart phones and computers. Hence, the security threats in MQTT protocol need to be identified to protect the IoT environment that is built on this protocol.

The remaining part of the paper is organised as: Section II contains background details of different application protocol and cyber attacks. Section III details about various attacks on MQTT protocol. Section IV security analysis like issues requirements Section V Different attacking scenarios on MQTT Protocol. Finally the conclusion is presented in last Section.

II. BACKGROUND

2.1 APPLICATION LAYER PROTOCOLS IN IOT

The advancement in mobile devices and communication systems provided a rapid momentum for IoT based solutions. With the rapid increase in overall utilization of IoT enabled systems, there is a rapid growth in the amount of data captured [10]. Moreover, to create more valuable services, data produced needs to be managed according to its requirements. Thus, integration of WSN or IoT with cloud computing is becoming very essential. Figure 1 depicts the generic model of communication and data exchange between IoT and external storage sources like Cloud and HPC cluster. The figure indicates several of application layer protocols used in the communications. Keeping the characteristics of IoT devices such as the large area deployment, minimal power consumption, and weak computational capabilities, there is a great need for efficient application layer protocols. They must be well-tuned for seamless data transfer from IoT devices to the external sources. These protocols are expected to have some properties. 1) It is necessary to fit the protocol object code into the little memory of IoT devices. 2) The p rotocols must not consume too much power to process data and send it to the external source. 3) The protocols must be suitable for interoperable environments. To that extent, they need to follow the standards. 4) The protocols must be open source to get widespread acceptance and usability.

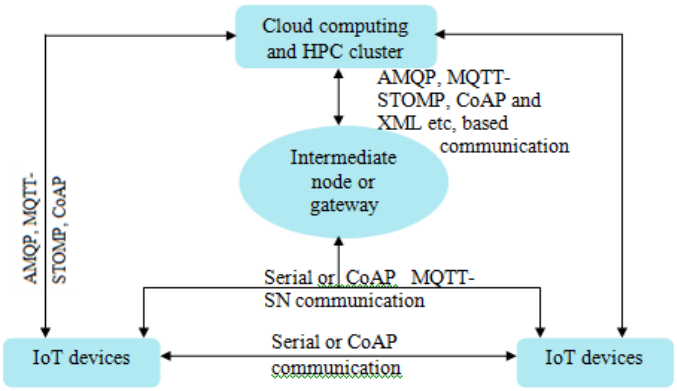

Figure 1.IoT Devices and cloud Communication Model

To provide information exchange services to a wide group of IoT devices efficiently, application protocols must deliver services in a publisher/subscriber model which is simple to operate. The client who wants data must register service request through a broker and is popularly known as a subscriber. In the same way, some clients constantly produce data; they must register as a publisher with the broker. The publisher/subscriber models work on the paradigm that publisher constantly produces the data/service and subscriber constantly consumes the data/service. The overall architecture is straightforward and easy to implement. Some of the well-known data exchange protocols are: Message Queuing Telemetry Transport [11], Constrained Application Protocol [12], Extensible Messaging and Presence Protocol [13], Data Distribution Service, Advanced Message Queuing Protocol [15]. Out of these protocols, MQTT, AMQP, and COAP are widely used. These protocols are discussed in detail in the following subsections [27].

A. Advanced Message Queuing Protocol

This protocol is highly useful for the exchange of bulk messages. Using this protocol, massive amounts of data can be sent securely to the data store without considering the performance of the whole system [10]. It is an open standard for Message Oriented Middleware (MOM) communication. AMQP middleware supports different applications to send and receive messages irrespective of platform and different programming languages used. In AMQP, the messages are self-contained, and data contained in a message is not transparent and unchangeable. AMQP provides various ways to transfer messages such as Publish-and subscribe , store-and-forward and point-to-point.

An AMQP messaging system consists of four main components: 1) Publisher: An application that generates messages and sends it to the AMQP broker. 2) Consumer/Subscriber: An application that receives the messages from one or more publishers through a broker. 3) Broker/Server: A server or a daemon program that contains one or more Message Queues, Virtual Hosts, Exchanges, and Bindings. 4) Virtual Host: A daemon application program, having services such as Hosts, Exchanges, Message Queues, and Bindings. Each of these components run on autonomous hosts or machines.

B. Constrained Application Protocol

This protocol is specially designed for machine-to-machine applications[10]. The CoAP Protocol provides a request/response interaction model. It supports the built-in discovery of services and resources, provides a feature like asynchronous message exchanging, and includes key concepts of the Web such as URIs. This protocol uses the Datagram Transport Layer Security as the security protocol for authentication and Integrity. The CoAP protocol consists of some main components, they are: Endpoint: It can either be source or destination of a CoAP message. It is identified by IP address and a UDP port number. Sender: The source's endpoint that generates messages. Recipient: The destination's endpoint that consumes messages. Client: A client is similar to a sender or recipient, but it can place a request instead of the message. Server: a server is a receiver at the destination endpoint of a request; It provides response to request. Origin Server: The resource are resides on origin server.

Intermediary: An endpoint that acts as both server and a client towards an origin server. A common form of an intermediary is a proxy. Proxy: It helps in packet forwarding, provides namespace support, and also does protocol translation.

C. Extensible Messaging and Presence Protocol

The XMPP is an open standard protocol. The purpose of this protocol is to transmit numerous minor codes of XML data over a decentralized network in real time. [16] XMPP was first introduced in late 1990s by Jabber. It was used for instant messaging. It can also be used for wide range of applications besides instant messaging, including multi-party chat, voice & video calls, collaboration, and general routing of XML data. XMPP-IoT version allows sending and receiving of messages between machines.

D. Data Distribution Service

DDS protocol is a middleware protocol for data-centric connectivity from the Object Management Group. DDS is used to integrates the components of a systems together and also this protocol providing low-latency data connectivity, scalable and extreme reliable architecture that business and mission-critical IoT application needs[16]. DDS protocol used for different applications that require real-time data exchange. DDS needed for different applications like defense and aerospace, air-traffic control, autonomous vehicles, medical devices, power generation, robotics, simulation and testing, smart grid management, transportation systems, and other applications that required for the real-time data exchange.

E. Message Queuing Telemetry Transport (MQTT)

MQTT uses the client/server model. [27] Every device that is connected to a server, using TCP known as (broker) message in MQTT is a discrete chunk of data and it is ambiguous for the broker[17]. Therefore, MQTT is a message oriented protocol. The address that the message published to it is called topic. The Device may subscribe to more than one topics, and it receives all messages that are published to these topics. The broker is a central device between the spoke model and the mentioned hub. The main MQTT broker responsibilities are processing the communication between MQTT clients and distributing the messages between them based on their interested topics. The broker can deal with thousands of connected devices at the same time. Upon receiving the message, the broker must search and find all the devices that own a subscription to this topic. MQTT client may be any of IoT object that sends or receive data, not just devices. Any device can be a client (e,g, its role in the system whether it is a subscriber or a publisher. MQTT Broker is a central device between the spoke model and the mentioned hub. The main MQTT broker responsibilities are processing the communication between MQTT clients and distributing the messages between them based on their interested topics. The broker can deal with thousands of connected devices at the same time. Upon receiving the message, the broker must search and find all the devices that own a subscription to this topic.

MQTT architecture that contains three components. Those are a publisher, a broker, and a subscriber. The device that is interested in a specific topics registers on it as a subscriber to be informed when the publishers publishing his topics by the broker. The publisher transfers the information to the subscribers via the broker. It is working as a generator of interested data, then, the authorization of the subscribers and the publishers are checked by the broker to realize the associated security issues. Figure 2 presents the component of the MQTT architecture.

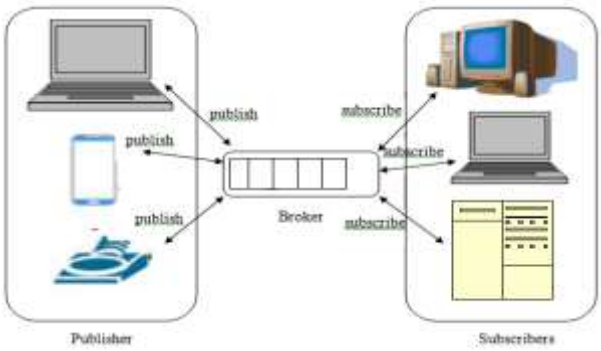

Figure 2.The Architecture of MQTT Protocol

When compared to other protocols like HTTP and other IoT messaging protocol, the Message Queuing Telemetry Transport protocol has a considerably smaller footprint, making this protocol much more suitable for resource-constrained environments. It has many advantages, every MQTT-based broker has different or comparable abilities for entity encryption or authentication [9]. The MQTT protocol was designed specifically for asynchronous communication, where subscriptions to entities or publishing from entities take place in a parallel order. The protocol is also able to provide reliable transfers by choosing different reliability mechanism, also called Quality of Service. MQTT uses the publish-subscribe communication model, where the client themselves do not require updates, thus in turn causing reduction of used resources, which makes this model optimal for use in a low-bandwidth environment.

2.2 COMMON CYBER ATTACKS

A. Denial-of-service attack

A DoS attack is a type of cyber attack in which a malicious actor makes a computer or other device unavailable to its intended users by interrupting the device's normal functioning. [21]This attacks are carried out by overwhelming or flooding a targeted

machine with requests until normal traffic is unable to be processed. A DoS attack is performed by using a single computer to launch the attack. The main focus of a DoS attack is to overwhelm the capacity of a targeted machine, resulting in denial-of-service to additional requests. The multiple attack vectors of DoS attacks can be grouped by their similarities. Figure 3 Presents DoS attack.

Figure 3.Denial of Service attack

B. Distributed Denial-of-service attack

A DDoS attack is a malicious attack which attempt to distort normal visitor of a targeted server, carrier or network through its surrounding infrastructure with a flood of Internet site visitors [22].This attacks are carried by utilizing multiple compromised computer systems as sources of attack traffic. Exploited machines can confine computer system and different networked resources such as IoT devices. Figure 4 presents Distributed Denial of Service attack.

Figure 4. Distributed Denial of Service attack

C. Man-in-the-middle attack (MitM)

Figure 5 Shows a MitM attack, that is carried out by the hackers to interfere between communications of two parties to get all information communicated between them. [23] The hacker can stop data transfer between sender and receiver or can redirect the same messages to another user. The MitM attack intrudes the user's communication, hiding their presence, and making it appear normal as if there is no third party involved in the communication. The main aim of hacker is to steal user login information, credit card details and financial details and so on. [24] It is important to take safety measures to prevent MitM attacks before they occur rather than trying to detect them while they are actively occurring. There are some attacking techniques to prevent MITM attacks from compromising your communications. They are Sniffing, Packet Injection, Session Hijacking and SSL Stripping.

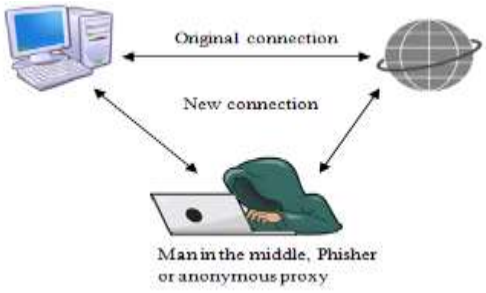

Figure 5.Man in the middle Attack

D. Phishing

Phishing is a type of social engineering attack which is used to get user data that include login credential and credit card numbers. This attack happens when attacker is pretending as a trusted entity and makes a victim into opening an email, textual message or instant message. The receiver is tricked to click a malicious link which result in the installation of malware, ransomware assault which makes system unusable or disclose of sensitive information. In Figure 6 Phishing attempts most customarily begin with an email attempting to achieve sensitive information through a few user interactions, which includes clicking on a malicious link or downloading an inflamed attachment. Phishing scams not only use the link, it can also employ text messages, phone calls and social media tools to make victims to provide sensitive information [19].

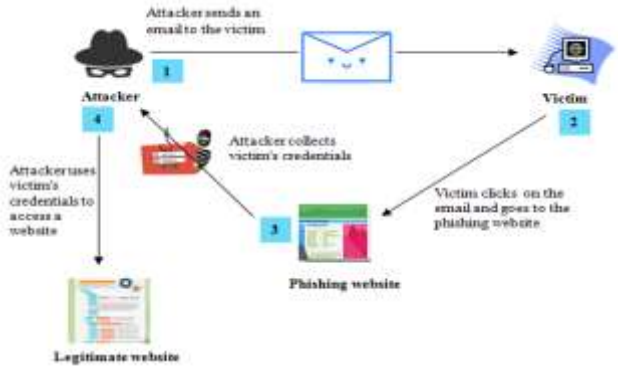

Figure 6.Phishing Attack

E. Intrusion

A network intrusion is an unauthorized activity on a computer network. Intrusion can be detected based on the understanding of how attacks work. In maximum of the cases, this type of unwanted activities absorbs resources of network considered for other uses, and nearly continually threatens the safety of the network and/or its data. Figure 7 shows Network intrusion detection system will help to block the intruders [4].

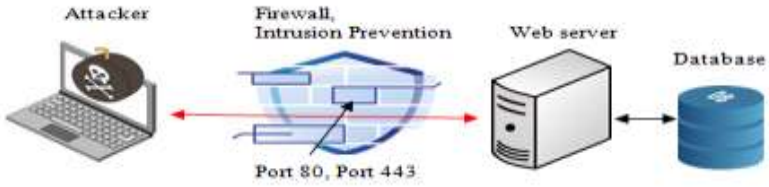

Figure 7.Intrusion Attack

III. ATTACKS ON MQTT PROTOCOL

Because of its simplicity and scalability, MQTT is generally used application layer protocol to transfer data among many IoT devices, Compared to all other protocols [5]. MQTT was designed for light-weight communications between constrained resource devices such as mobile phones and servers. Figure 8 shows general MQTT Protocol Publish-Subscribe model [6]. In this pattern, publisher, subscriber, and broker are the basic elements for establishing communication among many devices in IoT [21]. This protocol follows TCP-based connection establishment process. At first, publisher device sends request message i.e., CONNECT, to connect with the broker. After the request is received by the broker, the broker will send the acknowledgment, CONNACK, to the publisher device. After receiving acknowledgment from the broker, the publisher device sends or publishes the message on a specific topic to the broker, and finally the receiving devices subscribe the messages from the broker.

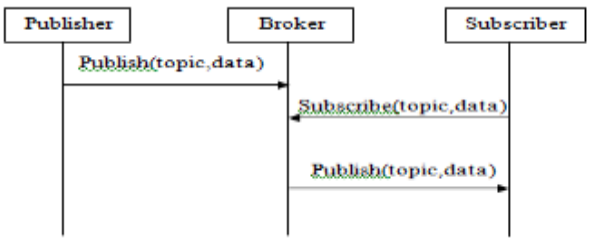

Figure 8. MQTT Protocol Publish-Subscribe model

There are several attacks against the MQTT protocol in the test environment. The malicious devices get an access to the network and it prevents the services proffered by the broker during publishing and subscribing the messages. By analyzing publish-subscribe messages we can identify the attacks against the MQTT broker. Here these attacks are at the network level along with all generated traffic. The attacks carried out were as follows:

A. Denial-of-service attack

The DoS attack scenario shown in Figure 9.The MQTT broker has to be analyzed by receiving the network traffic. An attacker can initiate a DoS attack in the broker by frequently sending multiple connection requests hence making the broker busy as in flooding attack. If multiple connection requests reach at the same time, then the buffer will be drained and the broker will not be able to handle all new incoming requests. Also the broker is not able to differentiate between normal and hoax CONNECT message packets [5]. When the broker receives flood request messages, it starts to acknowledge with CONNACK message.

During DoS attack, there is a rapid rate of increase in the number of CONNECT and CONNACK packets. Which is leads to halt the broker service and prohibits the functioning of the intended IoT network.

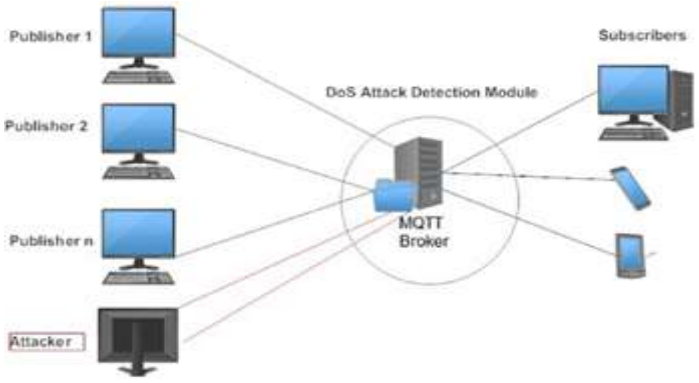

Figure 9.DoS attack scenario in MQTT

B. Man-in-the-middle attack.

MitM interrupt the messages communicate between two points to modify the content, this is done between a broker and the sensor by modifying the sensor data. Attack accomplishing tools are distribution Kali Linux and the tool Ettercap [30].

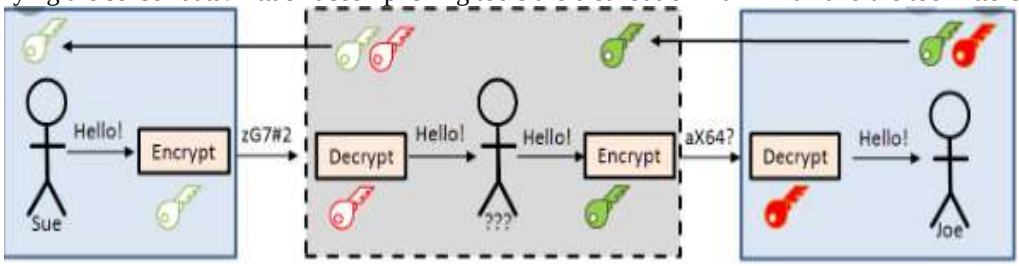

Figure 10 Man-in-the-middle attack in MQTT

Figure 10 shows MitM attack in MQTT protocol. MQTT was designed for light-weight communications between constrained resource devices such as mobile phones and servers. Although security was not designed into the protocol, it provides some security safeguards [28].This protocol enables a two-way hand shake by allowing client authentication. If SSL/TLS is available on the constrained resource devices then this mechanism allows for encryption of data in the message. When SSL/TLS is not available, the user name and password that authenticate the client are in the clear instances. This two-way handshake is vulnerable to man-in-the-middle attacks. Both mutual authentication and encryption are needed to avoid MitM attacks.

C. Intrusion

A network intrusion is an unauthorized activity on a computer network. Intrusion is detected based on the defenders having a clear understanding of how attack can works [14]. In some cases, such an unused activity uses network resources for other uses, and nearly always threatens the safety of the network or its data or both. Properly designing and deploying a network IDS (intrusion detection system) will help block the intruders. Intrusion taking under consideration the characteristics of the MQTT protocol. This attack consists of using the well-known port for this protocol and a command that uses the special character "#" are often employed by an external attacker for knowing the active topics available for being subscripted [31].

IV. SECURITY ANALYSIS ON MQTT

A. Security Overview

An attack can cause damage to the user in different domains. MQTT provides different security mechanisms in that many of them are not configured like data encryption or entity authentication. During authentication mechanisms, when device tries to connect with broker, the broker registers device information that includes physical address of device (MAC). And broker can do access authorization using Access Control List (ACL). The ACL includes information such as password and identifier of the different clients, which allows accessing different objects and can also specify the client which function it needs to be perform.

Confidentiality is a major requirement for security system. This can be achieved at the application layer by encrypting message at publisher side. This type of encryption can be achieved as either end to end or client to broker model. In client to broker type, broker decrypts the information that comes from publisher and encrypts the information that needs to be forwarded to other side client. In end to end type, broker cannot decrypt the information comes from publisher instead of that it directly forwards cipher text to other device. In other methods broker does not requires any additional requirements for encrypt/decrypt messages except few computational resources and less energy.

B. Security Requirement and Attacking Surfaces

To select protocol for IoT devices, data security is most important constraint to be considered because some of IoT communication protocol does not have data security mechanism [3]. The Data security is made up of three main components: data confidentiality, data availability and data integrity. It also contains additional security requirement to provide access like authentication, authorization. MQTT protocol does not have complete security mechanism; it contains only authentication mechanism without encryption capabilities [19].

While developing applications IoT developer has to take some considerations to design solution for security in the IoT communication protocol. They are 1) IoT device requires a lightweight security protocol because of limitations in IoT devices. 2) Each of connected IoT devices uses different protocol and different security mechanism in heterogeneous environment. 3) The reliability of network might force us to use minimum overhead in security mechanism .

While considering the requirement for security we need to concentrate on attack surface in IoT. [20] The IoT's attack surface in is divided into Public network and local network. The local network is called as internal attack, here the attacker and IoT devices are in the same network while the public network is also called as external attack, here the attacker might present anywhere in the public network to attack the IoT system.

C. Reasons why IoT implementation does not use security mechanism.

1. Resource Constrained Device

One of the main reasons not to use security mechanism is limitations in IoT devices. Many devices are categorized based on RFC 7228 as constrained device [25], these devices are further classified into three classes based on their RAM data size and ROM code size as follows.

Table - 1 Classes in Constraint Device - RFC 7228

	Flash (Code Size)	**RAM(Data Size)**
Class 0	**<<100KB**	**<<10KB**
Class 1	**~100KB**	**~10KB**
Class 2	**~250KB**	**~50KB**

Most of the resource constrained devices has very limited computing performance specially class 0 devices which cannot handle many security approaches [15] particularly the mechanism which has heavy computation.

2. Lack of security awareness

IT and other organizations need to improve the awareness of IoT treats. Lack of knowledge on security leads to increase the challenges in security of connected devices also increases treat level, which keeps organization at risk. Few key capabilities are considered by IT and security decision makers to protect against security attacks in IoT.

3. Huge number of devices

Extensive number of devices are connected in IoT. More vulnerability is created by number of connected devices. In IT department, it is necessary to manage large number of different types of devices when IoT system is applied with security mechanism [26]. For instance, if IT department gives authentication by using username and password then they have to put lot of effort to maintain security credentials.

V. ATTACK SCENARIOS ON MQTT PROTOCOL

How attack can be attacked on MQTT protocol can be explained in this part of the paper. Initially attacker had no idea about the victim system that they want to attack, like no idea about communication channel, infrastructure and defense mechanism[18]. Attack can be begins by collecting the related information by using Massca or Shodan search engine. This paper uses Shodan search engine to collect related information to attack on MQTT protocol. Attacker can use port number 1883 to search about MQTT protocol. Attacker has to type "port: 1883 "MQTT" " inside of search box of Shodan. Port number 1883 is the default port for MQTT broker. Finally Search result shown in the Figure 11. This shows 24998 brokers with default port successfully displayed on Shodan page.

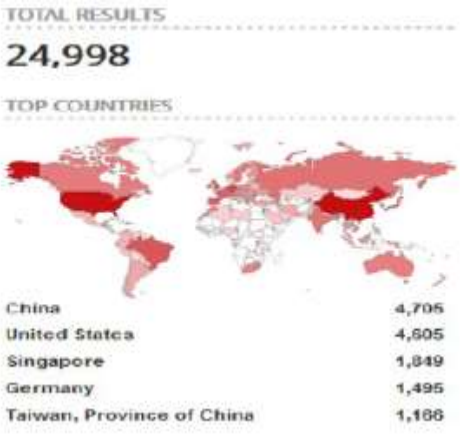

Figure 11. Result of MQTT broker on port 1883 in Shodan

The Figure 12 represents the MQTT connection code. If brokers have connection code "0" it indicates that the broker does not use any type of client authentication mechanism so unknown subscriber or publisher can easily connect with broker. In Figure 12 all broker have code "0", thus all can be easily attacked by attacker.

Figure 12. MQTT connection code in Shodan Page search result

First scenario illustrated in Figure 13.When all broker connection code is "0", attacker start subscribing to all topics in broker (subscribe to #) which may give sensitive information that is to be analyzed later.

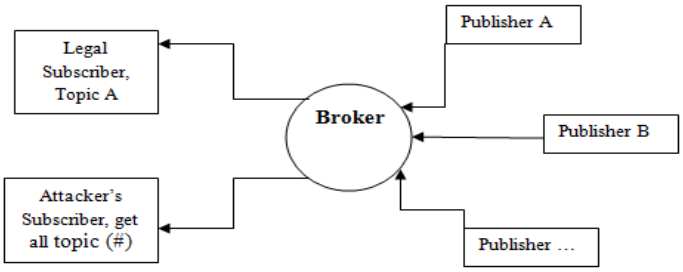

Figure 13.First scenario

Another scenario explained in Figure 14. Here attack can be initiated by publishing information to the broker. In this scenario subscriber are street lamps, to control street lamps legal publisher has to publish a message. On another side attacker can get a related messages to control street lamps by subscribing to broker. By analyzing the control message, the attacker can easily publish his message. The attacker can utilize this type of scenario to publish spam information.

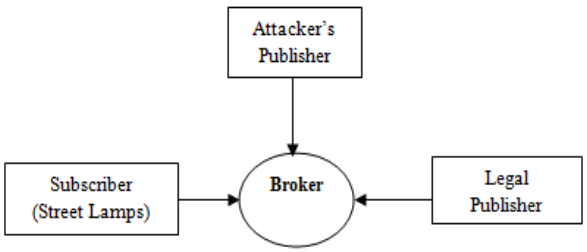

Figure 14.Scenario two

Two scenarios discussed above are common scenarios that can affect on both and public and local network. Next scenario assumes that attacker is connected to same network with IoT system. Based on this assumption attacker can get related information by analyzing traffic on network while data in-transit. Collected information is in the form of plain text, like name of topic, port number, IP broker and data payload of MQTT used in IoT system. An attack can be carried out by using the Wireshark and Ettercap. [18]The attacker and publisher are in same network, publisher can detect and modify the data that is in transit therefore attacker can accomplish the data authentication, data privacy and data integrity of MQTT packets.

A. Data privacy

MQTT protocol does not provide any data encryption by default, data privacy in MQTT message is an issue. Due to this reason whether authentication mechanism used by the broker or not, the attacker can still detect the data in transit easily. Figure 15 represents the data transit from publisher device captured by Wireshark. This data includes message and topic of MQTT.

Figure 15. MQTT topic and message captured in Wireshark

B. Authentication

If broker uses client authentication mechanism, then it needs to use username and password to get an authentication. Attacker cannot play publisher or subscriber role if he does not know username and password. In above scenario both attacker and publisher are in the same network.Therefore attacker can easily sniff the traffic on this network. While CONNECT packet transit from publisher to broker to get connect with Broker, the username and password is revealed thus attacker can easily attack. Figure 16 shows the CONNECT packet transit from the publisher to broker that has been sniffed by the attacker.

In Authentication Process, CONNECT packet includes header i.e., KeepAlive which tells how long IoT device connect with broker. Thus when header time expired the device again sends the CONNECT packet to broker to restart connection.

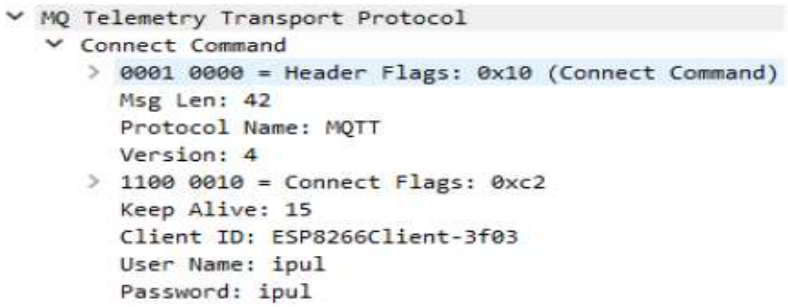

Figure 16.The MQTT Connect command packet

C. Data integrity

Another type of attack is data integrity this is mainly targeting on integrity of data in transit. In this attack, attacker already knows about data packets that can be modified during transmission. In this scenario attacker wants to change the topic name from “outTopic” to “outTopuc”. To change the topic name, attacker creates a filter file named as owned.filter which filter data packet in transit that has broker IP as destination address and TCP port 1883. When packet identifies the matched filter criteria, it starts searching for “outTopic” and changes it with “outTopuc” as shown in Figure 17. After changing topic name, Etterfilter application is used for compile the filter file which finally gives an output file “owned.ef”.

```
#owned.filter
if (ip.proto == TCP && tcp.dst == 1883 && ip.dst == 'IP Broker' &&
search(DATA.data, "outTopic")) {
  replace("outTopic", "outTopuc");
  msg("payload replaced\n");
}
```

Figure 17.Replacing topic name of MQTT data packet

Ettercap is an free and open source tool or application which run on specific interface, the attacker use this interface to connect to internet and attacker uses compiled filter file to modify the packet. This step is given in Figure 18.

```
etterfilter owned.filter -o owned.ef

ettercap -T -q -i eth0 -F owned.ef -M ARP /// ///
```

Figure 18. Command to run ettercap application on specific interface

After successfully change the published message topic name that is received by subscriber device. This is represented in Figure 19.

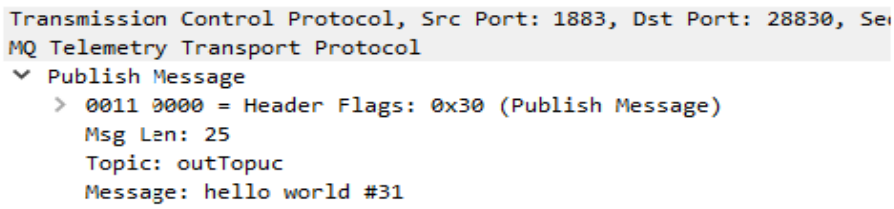

Figure 19. Result for change in topic name

VI. CONCLUSION

MQTT is a widely used application protocol in IoT system. Providing security to MQTT protocol is very important compare to all other protocols because of its simplicity and scalability. The summary of this paper tells that it is necessary to protect the IoT connected devices from malicious attacks and misuse which could prevent the evolution of IoT as a secure and reliable paradigm. Before providing security need to know about what are the different security scenarios, This paper gives idea about different attacking scenarios to detect different attacks that target the IoT connected devices in MQTT protocol. Also discussed about the requirements to provide security for MQTT protocol.

REFERENCES

[1] Syed Naeem Firdous, Zubair Baig, Craig Valli and Ahmed Ibrahim "Modeling and Evaluation of Malicious Attacks against the IoT MQTT Protocol", 2017 IEEE International Conference on Internet of Things, 2017.

[2] Ahmad W. Atamli and Andrew Martin,"Threat-based Security Analysis for the Internet of Things", International Workshop on Secure Internet of Things, 2014.

[3] Syaiful Andy, Budi Rahardjo and Bagus Hanindhito, "Attack Scenarios and Security Analysis of MQTT Communication Protocol in IoT System", Proc. EECSI 2017, Yogyakarta, Indonesia, 19-21 September 2017.

[4]Robert Moskowitz, Network Intrusion: Methods of Attack, [Online], Available: http://www.rsaconference.com/industry-topics/blog/network-intrusion-methods-of-attack

[5] Haripriya A. P and Kulothungan K, "Secure-MQTT: an efficient fuzzy logic-based approach to detect DoS attack in MQTT protocol for internet of things", A. P. and K. EURASIP Journal on Wireless Communications and Networking, 2019.

[6] Juanita Koilpillai, is MQTT is secure for IoT, [Online], Available: https://www.waverleylabs.com/is-mqtt-secure-for-the-iot-only-with-an-sdp/

[7] Tarfa Hamed, Jason B. Ernst, and Stefan C. Kremer, "A Survey and Taxonomy of Classifiers of Intrusion Detection Systems", Springer International Publishing AG 2018.

[8] Lane, T. (2006). A decision-theoritic, semi-supervised model for intrusion detection. In Machine learning and data mining for computer security, pp. 157–177. London: Springer.

[9]Dan Dinculeana and Xiaochun Cheng, "Vulnerabilities and Limitations of MQTT Protocol Used between IoT Devices", Applied Science 2019, 9, 848, 2019.

[10] Ajay Chaudhary, Sateesh K. Peddoju, and Kavitha Kadarla ,"Study of Internet-of-Things Messaging Protocols used for Exchanging Data with External Sources", 2017 IEEE 14th International Conference on Mobile Ad Hoc and Sensor Systems, 2017.

[11] A. Banks and R. Gupta, "Mqtt version 3.1. 1," OASIS standard, vol. 29, 2014.

[12] Z. Shelby, K. Hartke, and C. Bormann, "The constrained application protocol (coap)," 2014.

[13] (2017) XMPP protocol specification. [Online]. Available: http://xmpp.org/

[14] Hector Alaiz-Moreton and Jose Aveleira-Mata, "Multiclass Classification Procedure for Detecting Attacks on MQTT-IoT Protocol" in Research Article, 2019.

[15] (2017) AMQP protocol specification. [Online]. Available: https://www.amqp.org/

[16] XMPP-protocol,[Online]. Available: https://www.element14.com/community/docs/DOC-92109/l/tech-spotlight-the-xmpp-protocol

[17] Muneer Bani Yassein, Mohammed Q. Shatnawi and Shadi Aljwarneh, Razan Al-Hatmi, "Internet of Things: Survey and open issues of MQTT Protocol", ICEMIS2017, Monastir, Tunisia, 2017.

[18] S. Andy, B. Rahardjo, and B. Hanindhito, "Attack Scenarios and Security Analysis of MQTT Communication Protocol in IoT System," in Proceedings of the 4th International Conference on Electrical Engineering, Computer Science and Informatics, EECSI 2017, pp. 19–21, IEEE, Yogyakarta, Indonesia, 2017.

[19] ISACA Volunteer Member, "Cybersecurity Fundamentals Study Guide," ISACA, 2015.

[20] M. M. Hossain, M. Fotouhi and R. Hasan, "Towards an Analysis of Security Issues, Challenges, and Open Problems in the Internet of Things," 2015 IEEE World Congress on Services, New York City, NY, 2015, pp. 21-28.

[21]Denial-of-Service, [Online], Available:https://www.cloudflare.com/learning/ddos/glossary/denial-of-service/

[22]Disributed Denial-of-Service, [Online], Available: https://www.cloudflare.com/en-in/learning/ddos/what-is-a-ddos-attack/

[23] Man in the Middle attack, [Online], Available: https://securebox.comodo.com/ssl-sniffing/man-in-the-middle-attack/

[24]Types of Man in the Middle attack, [Online], Available: https://www.rapid7.com/fundamentals/man-in-the-middle-attacks/

[25] C. Bormann, M. Ersue, and A. Keranen, " RFC 7228 Terminology for Constrained-Node Networks," IETF, May 2014.

[26] Iot Security Awareness. InfoSec Institute [Online]. Available: http://resources.infosecinstitute.com/iot-security-awareness/

[27]Bhanujyothi H C, Rajesh S M, Vidya J and Sahana D S, "A Study on IoT Messaging Protocols and it's Comparision for implementation of IoT Services",in International Journal of Scientific and Research Publications 2019,Volume 9, Issue 3, pp. 596-601.