SECURE HYBRID SCHEME FOR SECURING MQTT PROTOCOL BASED ON ENHANCED SYMMETRIC ALGORITHM

AHMED JAMEEL HINTAW

UNIVERSITI SAINS MALAYSIA

2023

SECURE HYBRID SCHEME FOR SECURING MQTT PROTOCOL BASED ON ENHANCED SYMMETRIC ALGORITHM

by

AHMED JAMEEL HINTAW

Thesis submitted in fulfilment of the requirements
for the degree of
Doctor of Philosophy

March 2023

ACKNOWLEDGEMENT

Encouragement, inspiration, support, and the assistance of others are how great work and impossible tasks are ever accomplished. I have been lucky enough to have many people in my life who believe in and respect other accomplishments and are willing to guide and direct them to achieve their goals. I would like to give special thanks to those people who believed in me to accomplish my Ph.D. journey.

First and foremost, I would like to acknowledge my main supervisor, Associate Professor Dr. Selvakumar A/L Manickam. I am indebted to him for being more than my mentor, but an inspiration for me. At every major point in the process, I was amazed at Associate Professor Dr. Selvakumar A/L Manickam innate ability to anticipate my concerns and needs and to skillfully guide me in the direction I needed to go. Also, special thanks to my co- supervisor TS. Dr. Shankar A/L Karuppayah as well as my field supervisor Dr. Mohammed Faiz Abomaali for their wisdom and willingness to assist and guide me in this Ph.D. journey. Next, my sincere appreciation and thanks are dedicated to all my respondents for their valuable feedback.

To my parents and family members, my father, my mother, my wife Dr. Noor Alkharsain, and my angel daughter Dan would like to express my love and deep appreciation for their patience and moral support, specifically my dear brothers Khalid Hintaw, Ameer Hintaw, and my sisters, Dr. Noor Hintaw, Maryam Hintaw, who made sacrifices along the way to help me complete this Ph.D. journey on time.

Finally, I would like to thank friends and all those who have rendered their help directly or indirectly throughout this Ph.D. journey. Their encouragement, assistance, and support are highly appreciated.

TABLE OF CONTENTS

ACKNOWLEDGEMENT ..... **ii**

TABLE OF CONTENTS ..... **iii**

LIST OF TABLES ..... **ix**

LIST OF FIGURES ..... **xi**

LIST OF SYMBOLS ..... **xiii**

LIST OF ABBREVIATIONS ..... **xv**

ABSTRAK ..... **xviii**

ABSTRACT ..... **xx**

CHAPTER 1 INTRODUCTION **1**

1.1 Introduction 1

1.2 IoT Security 4

1.2.1 Identify IoT Security Goals and Security Attack 6

1.2.2 IoT Data Protocols 7

1.3 Research Motivation 10

1.4 Problem Statement 11

1.5 Research Goal and Hypothesis 13

1.6 Research Questions and Objectives 14

1.7 Research Contributions 15

1.8 Research Scope and Limitation 16

1.9 Research Steps 17

1.10 Thesis Organization 18

CHAPTER 2 LITERATURE REVIEW **20**

2.1 Overview of IoT 20

2.1.1 IoT Infrastructure 21

2.2 Underlying Concept About The MQTT Protocol 23

2.2.1 Message Queuing Telemetry Protocol 23
2.2.2 Technical Aspects of the MQTT Protocol 25
2.2.2(a) Publish/Subscribe 25
2.2.2(b) Topics and Subscriptions 26
2.2.2(c) Quality of Service Levels 26
2.2.2(d) Retained Messages 27
2.2.2(e) Clean Sessions and Reliable Connections 27
2.2.2(f) Wills 27
2.3 MQTT SECURITY 28
2.4 MQTT THREAT MODEL 30
2.4.1 Threat Actors 30
2.4.2 Threats and Impact 31
2.5 MQTT Attacks Taxonomy and Countermeasures 35
2.5.1 TCP-Based Attacks 35
2.5.2 MQTT Protocol-Based Attacks 37
2.5.3 TLS-Based Attacks 38
2.5.4 Data at Rest-Based Attacks 39
2.6 Protection Levels of MQTT Protocol from Various Security Threats 44
2.6.1 Network Layer Security 45
2.6.2 Transport Layer Security 45
2.6.3 Application Layer Security 46
2.6.4 Physical Layer Security 47
2.7 Related Works on Securing MQTT Protocol 47
2.7.1 Confidentiality 49
2.7.1(a) Asymmetric Scheme 50
2.7.1(b) Symmetric Scheme 52
2.7.1(c) Hybrid Scheme 55

2.7.2 Access Control 57
2.7.3 Summary of Related Works 58
2.8 Possible Solutions 62
2.8.1 Machine Learning 62
2.8.2 Blockchain Technology 63
2.8.3 Artificial Intelligence 64
2.8.4 Ciphering 64
2.8.4(a) Cryptanalysis 66
2.8.4(b) Dynamic Encryption 69
2.9 New Security Scheme Requirements 73
2.9.1 Lightweight and Secure 73
2.9.2 Integrated Scheme 74
2.9.3 Self-Controlled Scheme 74
2.10 Chapter Summary 75
CHAPTER 3 METHODOLOGY 76
3.1 The General Structure of the Proposed Scheme 77
3.1.1 Set-up Phase (Key Authority) 77
3.1.2 MQTT Hybrid Encryption Phase 78
3.1.3 MQTT Hybrid Decryption Phase 78
3.2 Methodology of Designed Hybrid Scheme 79
3.2.1 Assumptions 79
3.2.2 Design Goal 79
3.3 Overview of the Proposed Scheme 81
3.3.1 Secure Hybrid Scheme 83
3.3.2 Proposed Symmetric Algorithm 83
3.3.2(a) SubBytes Transformation 84
3.3.2(b) ShiftRows Transformation 85

3.3.2(c) MixColumns Transformation 86
3.3.2(d) AddRoundKey Transformation 87
3.3.2(e) Key Expansion Module 87
3.3.3 Proposed D-AES Algorithm with Enhanced Security 88
3.3.3(a) Enhanced Key Expansion 90
3.3.3(b) Enhanced SubBytes Transformation 93
3.3.3(c) Enhanced InvSubBytes Transformation 95
3.3.3(d) Enhanced ShiftRows Transformation 97
3.3.3(e) Enhanced InvShiftRows Transformation 100
3.3.4 Designed Secure Hybrid Scheme 102
3.3.5 Broadcast Encryption 106
3.4 Chapter Summary 107
CHAPTER 4 IMPLEMENTATION OF THE PROPOSED SCHEME 108
4.1 Introduction 108
4.2 Proposed Scheme Requirements 109
4.2.1 Mosquitto Broker 109
4.2.2 Packet Analyzer Protocol 110
4.2.3 MQTT Test-Bed Environment Setup 110
4.2.4 Java Pairing Based Cryptography 112
4.3 Hardware and Software Specifications 112
4.3.1 Hardware Specifications 113
4.3.2 Software Specifications 113
4.4 Proposed Work Implementation Details 113
4.4.1 Proposed D-AES Implementation 114
4.4.1(a) D-AES Key Expansion Implementation 114
4.4.1(b) D-AES SubBytes Implementation 115
4.4.1(c) D-AES InvSubBytes Implementation 115

4.4.1(d) D-AES ShiftRows Implementation 116
4.4.1(e) D-AES InvShiftRows Implementation 116
4.4.2 MQTT Protocol Implementation 118
4.4.3 Secure Hybrid Scheme Implementation 119
4.4.3(a) Setup Phase (Key Authority) 121
4.4.3(b) Encryption Phase 123
4.4.3(c) Decryption Phase 124
4.5 Evaluation Metrics 129
4.5.1 Security Metrics 129
4.5.2 Performance Metrics 130
4.6 Chapter Summary 132
CHAPTER 5 EXPERIMENTAL RESULTS AND DISCUSSION 133
5.1 Introduction 133
5.2 Security Analysis of Enhanced Algorithm 134
5.2.1 Known Answer Test 134
5.2.2 Non-Linearity analysis 135
5.2.2(a) Balance 135
5.2.2(b) Strict Avalanche Criteria 136
5.2.2(c) Hamming Distance 138
5.3 Performance Analysis 142
5.3.1 Execution Time 142
5.3.1(a) Proposed Symmetric D-AES Algorithm 142
5.3.1(b) Secure Hybrid Scheme 144
5.3.2 Storage Overhead 149
5.3.3 Traffic Overhead 149
5.4 Comparative Analysis 151
5.4.1 Enhanced Symmetric D-AES Algorithm 152

5.4.1(a) Avalanche Effect 152
5.4.1(b) Processing Time 154
5.4.2 Secure Hybrid Scheme 156
5.4.2(a) Processing Time 156
5.4.2(b) Traffic Overhead 159
5.5 Resistance to Linear and Differential Attacks 160
5.6 Resistance to Side Channels Attacks 163
5.7 Passive Attacks 164
5.8 Chapter Summary 167
CHAPTER 6 CONCLUSION AND FUTURE WORKS 168
6.1 Introduction 168
6.2 Conclusion 168
6.3 Limitations and Future Research 171
REFERENCES 173
LIST OF PUBLICATIONS

LIST OF TABLES

Page

Table 1.1 IoT Arena Vulnerabilities and Their Responsible Weaknesses 5

Table 1.2 IoT Security Requirements 7

Table 1.3 Feature Comparison of The Main IoT Protocols 8

Table 1.4 Research Scope 16

Table 2.1 Some Solutions That Use The MQTT Protocol 24

Table 2.2 MQTT Operations Set. 28

Table 2.3 MQTT QoS Types 28

Table 2.4 MQTT Security Mechanism at Each Layer 30

Table 2.5 Different Attack Vectors That Can Accrue to The MQTT and Mitigation Methods 34

Table 2.6 TCP-Based Attacks With Associated Counteractions and Security Objectives 36

Table 2.7 Exhibits MQTT Threats With Their Impacted Security Objectives and Counteractions. 39

Table 2.8 Security Objects Achievements and Counteractions Related to The Data-Based Attack 42

Table 2.9 Security Analytical Comparisons of Different Attacks. 43

Table 2.10 MQTT Security Mechanism at Each Layer 47

Table 2.11 Summary of Related Works 58

Table 2.12 Promising Technologies That Address Security Issues to Secure MQTT 65

Table 2.13 The Architecture and Issues of The Existing Modification of The AES. 72

Table 3.1 Bite-Sized of The Byte Including Its Operations and Binary Value 99

Table 3.2 Association Between Pn and $Pn-1$ of The Enhanced Transformation. 99

Table 3.3 Bite-Sized Splitting of The Byte, Including Its Operations and Binary Values For The Invshift Transformation. 101

Table 4.1 The Specifications of The Required Hardware For Experimentations 113

Table 4.2 The Specifications of The Required Software For Experimentations 113

Table 5.1 Known Answer Test Vectors For Analysis. 135

Table 5.2 Balance Comparison of Enhanced Symmetric D-AES Algorithms. 136

Table 5.3 The Avalanche Effect (%) Comparison of The Ciphertext. 138

Table 5.4 Test Samples of The Round Analysis 139

Table 5.5 Avalanche Effect Comparison of The Internal Rounds 140

Table 5.6 Average Time Taken and Its Standard Deviation For The Standard Algorithm 143

Table 5.7 Average Time and Its Standard Deviation For the D-AES 143

Table 5.8 Standard MQTT: Processing Time (Milliseconds). 147

Table 5.9 Proposed Mechanism: Processing Time, With Overhead (Milliseconds). 147

Table 5.10 Comparison of The Avalanche Effect With The Existing Works 153

Table 5.11 The Comparison of Encryption Time With Existing Work. 155

Table 5.12 The Comparison of Decryption Time With Existing Work. 155

Table 5.13 Key Features of Existing Security Options For Securing MQTT 164

Table 6.1 The Mapping Research Outcome 171

LIST OF FIGURES

Page

Figure 1.1 Google Trends of MQTT, COAP, DPWS, and AMPP (Google, 2022). 7

Figure 1.2 Major Stages of The Research Process 17

Figure 2.1 Global IoT Market Size 2016 - 2026 (Ray, 2018). 21

Figure 2.2 Internet of Things Infrastructure (Mohanta et al., 2020). 22

Figure 2.3 MQTT Header 23

Figure 2.4 Publish-Subscribe Paradigm in The MQTT System 25

Figure 2.5 MQTT QoS Levels 27

Figure 2.6 Threat Actors in An MQTT System. 30

Figure 2.7 The Threat Model in An MQTT Environment. 32

Figure 2.8 Taxonomy of MQTT Protocol Attacks. 35

Figure 2.9 Protection Levels in MQTT Systems. 44

Figure 2.10 The Diagram of Related Works 49

Figure 3.1 The Mapping Diagram of The Research Problem, Objectives, and Contributions. 76

Figure 3.2 General Structure of The Proposed Scheme 77

Figure 3.3 The Block Diagram of The Proposed Secure Hybrid Scheme. 82

Figure 3.4 The AES Cryptosystem Block Diagram With a 128-Bit Key 84

Figure 3.5 The Architecture of The Proposed Key Expansion Block 93

Figure 3.6 The Decision Flowchart For The Enhanced Shiftrows Transformation. 100

Figure 3.7 The Block Diagram of The Enhanced D-AES Algorithm For 128-Bits 101

Figure 3.8 Secure Hybrid Scheme to Secure MQTT Protocol. 106

Figure 4.1 Test-Bed Environment Setup 111
Figure 4.2 MQTT Publish Message 118
Figure 4.3 MQTT Subscribe Message 119
Figure 4.4 Encrypted MQTT Publish Message With The Proposed Scheme .. 120
Figure 4.5 Secure Publisher-Subscriber Messages of MQTT Protocol in IoT.. 121
Figure 5.1 Strict Avalanche Criterion Comparison of The Ciphertext..... 137
Figure 5.2 Hamming Distance (%) of The Enhanced Algorithm. 140
Figure 5.3 Percentage of Avalanche Effect Between Expanded Keys of PT1.. 141
Figure 5.4 Percentage of Avalanche Effect Between Expanded Keys of PT2.. 142
Figure 5.5 Encryption Time: Existing Algorithm vs. Enhanced Algorithm. 144
Figure 5.6 Decryption Time: Existing Algorithm vs. Enhanced Algorithm. 144
Figure 5.7 Standard MQTT: Total Processing Time. 146
Figure 5.8 Proposed Mechanism: Processing Time. 146
Figure 5.9 Average Execution Time of The Hybrid Scheme for Securing MQTT. 148
Figure 5.10 Total Data Size of The Proposed Scheme in The MQTT Network . 150
Figure 5.11 Traffic Overhead of Proposed Secure Hybrid Scheme 151
Figure 5.12 Avalanche Effect Factor in Different AES. 152
Figure 5.13 Avalanche Effect Differences Comparison in Different Modifications 154
Figure 5.14 The Encryption Time With a Different Number of Attributes. 158
Figure 5.15 The Decryption Time With a Different Number of Attributes. 159

LIST OF SYMBOLS

F	An access tree
$attr$	Attribute
z	Children nodes in the KP-algorithm
C_{D-AES}	Ciphertext generated by the dynamic symmetric algorithm
$(wt_c(x))$	Column weight that represents the active columns in x
Sk_c	Cipher key
D_F	The decryption key for the access structure
D_F	The decryption key for the access structure
Enc	Encryption
a(y)	Fixed polynomial
K_{sv}	Four bytes vector of the key state
$KP-ABE-Dec$	KP-ABE decryption algorithm
E''	KP-ABE encryption
$KP-ABE-Enc$	KP-ABE encryption algorithm
min	Minimum
m	MQTT payload data
$S'_{m(i,j)}$	New state matrix
num_x	Number of children in the KeyGen algorithm
d_r	Points at random for the root node
$f(k), f(x), f(h)$	Polynomial functions within Galois Field 2^8 (GF (2^8))
q_x	A polynomial with a degree
P_n	Position Number
P_v	Position Value
p	Prime order for a bilinear group
PK_{kp}	Public parameters of the KP-ABE algorithm

g_2	Random element
s	Random number
r	Root node of the access structure tree
$R\,c\,on_k$	Round constant
$K_{r(i,j)}$	Round key
N_r	Round number of symmetric algorithm
S_k	Secret key of the dynamic symmetric algorithm
1^k	Security parameter
γ	Set of attributes
$S_{m(i,j)}$	State matrix
E'	Symmetric D-AES encryption
$S_{(i,j)k}$	Tetrad-based state matrix
Γ	The access tree
G_1	The bilinear group of prime order
e	The bilinear map
KP_m	The ciphertext of the KP-ABE algorithm
$H1$	The collision resistant function
ES_k	The encrypted private key of the D-AES
$\Delta_{i,s}$	The Lagrange coefficient
F	The linear transformation function
MK_{pk}	The master secret key of the KP-ABE algorithm
n	The maximum number of attributes
$wt(x)$	The vector x weight in D-AES
Sk_t	True key length
k_x	Value of the threshold

LIST OF ABBREVIATIONS

6LowPAN	IPv6 over Low-Power Wireless Personal Area Networks
API	Application Programming Interface
ABE	Attribute-based encryption
ACL	Access Control List
AES	Advanced Encryption Standard
AKC	Asymmetric Key Cryptosystem
API	Application Programming Interface
ARP	Address Resolution Protocol
ATA	Authorized Topic Access
AUPS	Authenticated Publish & Subscribe
CAVP	Cryptographic Algorithm Validation Program
CBC	Cipher Block Chaining
CIA	Confidentiality, Integrity, and Availability
CoAP	Constrained Application Protocol
CP-ABE	Ciphertext-Policy Attribute-Based Encryption
CT	Ciphertext
D2S	Devices and Server
D-AES	Dynamic Advanced Encryption Standard
DDoS	Distributed Denial of Service
DE	Dynamic Encryption
DNS	Domain Name System
DNS-SD	Domain Name System service discovery
DoS	Denial of Service
DPWS	Devices Profile for Web Services
DTLS	Datagram Transport Layer Security
DVR	Digital Video Recorder
ECC	Elliptic-Curve Cryptography
ECDH	Elliptic Curve Diffie-Hellman
FGAC	Fine-Grained Access Control
FIPS	Federal Information Processing Standards
GF	Galois Field

HSTS	HTTP Strict Transport Security
HTTP	Hypertext Transfer Protocol
IAS	Information Assurance and Security
IDS	Intrusion Detection System
IETF	Internet Engineering Task Force
IM	Instant Messaging
IoT	Internet of Things
IPv4	Internet Protocol Version 4
IPv6	Internet Protocol Version 6
jPBC	Java Pairing Based Cryptography
K=KA	Key Authority
KAT	Known Answer Test
KP-ABE	Key-Policy Attribute-Based Encryption
LLSEC	Link Layer Security
M2M	Machine-To-Machine
mDNS	Multicast Domain Name System
MEML	Message-Oriented Machine Learning Re-Training Protocol
MITM	Man-In-The Middle
MK	Master Key
MME	MQTT Message Encryption
MQTT	Message Queuing Telemetry Transport
MSB	Most Significant Bit
MSK	MQTT Session Key
NIST	National Institute of Standards and Technology
OASIS	Organization for the Advancement of Structured Information Standards
P2P	Peer-To-Peer
PC	Personal Computer
PIN	Personal Identification Number
PK	Public Parameters
PKI	Public Key Infrastructure
PM	Policy Management
PR	Propagation Ratio
PT	Plaintext
QoS	Quality of Service

RC	Research Contribution
RO	Research Objective
RCON	Round Constant
SAC	Strict Avalanche Criteria
SEA	Scalable Encryption Algorithm
SIT	Secure IoT
SKC	Symmetric Key Cryptosystem
SMC	Secure Multi-Cast
SMQTT	Secure Message Queuing Telemetry Transport
SOAP	Simple Object Access Protocol
SQL	Structured Query Language
SSL	Secure Sockets Layer
TCP	Transmission Control Protocol
TEA	Tiny Encryption Algorithm
TLS	Transport Layer Security
UCON	Usage Control
UTF-8	Unicode Transformation Format—8-Bit
UDP	User Datagram Protocol
VM	Virtual Machine
WSDL	Web Services Description Language
XML	Extensible Markup Language
XMPP	Extensible Messaging and Presence Protocol

SKEMA HIBRID SELAMAT UNTUK MELINDUNGI PROTOKOL MQTT BERASASKAN ALGORITMA SIMETRI YANG DIPERTINGKAT

ABSTRAK

Internet benda (IoT) membolehkan komunikasi peranti dan mesin menggunakan Protokol TCP/IP. Pengangkutan Telemetri Baris Gilir Mesej (MQTT) ialah protokol yang paling popular dan dijangka akan menjadi piawai de facto pemesejan IoT. Oleh itu, MQTT mesti mencapai tahap keselamatan yang cekap. Namun, kelemahan MQTT yang paling ketara ialah kekurangan mekanisme perlindungan yang hanya mengesahkan objek keselamatan mudah seperti pengesahan dan dasar pengesahan tanpa tersulit, bahkan langsung tiada mekanisme tersulit. Data boleh diubah oleh penceroboh semasa penghantaran. Ramai penyelidik yang telah mencadangkan pelbagai teknik keselamatan untuk menangani isu ini. Sementara itu, skema sedia ada untuk melindungi rangkaian MQTT menambah overhed pemprosesan pada peranti tetapi masih terdedah kepada pelbagai serangan. Oleh itu, penyelidikan ini membentangkan suatu skema bersepadu yang dikenali sebagai "Skema Hibrid Selamat" untuk melindungi protokol MQTT daripada sebarang eksploitasi yang mungkin menghasilkan serangan siber yang canggih. Sistem kripto yang dicadangkan menggunakan dua algoritma: varian dinamik Piawai Penyulitan Termaju (D-AES) dan Penyulitan Asas Atribut Polisi Kekunci (KP-ABE). Skema Hibrid Selamat memperkenalkan seni bina reka bentuk baharu bagi algoritma AES simetri untuk menyulitkan muatan MQTT yang dipanggil "D-AES." Unit pengembangan kekunci telah diperkukuh dalam D-AES. Selain itu, nilai parameter algoritma berubah dengan setiap kekunci baharu semasa pelaksanaan algoritma. Oleh itu, polimorfisme telah dicapai, manakala sifat saling kendali dapat dikekalkan. Telah ditunjukkan bahawa

sifer tidak boleh ditembusi menggunakan analisis kriptografi pembezaan dan linear. Selain itu, bahagian kedua sistem kriptografi hibrid yang dicadangkan ialah KP-ABE, yang digunakan untuk mensifer kekunci persendirian D-AES untuk mengelakkan overhed pengiraan peta bilinear. Untuk menjalankan eksperimen, satu tapak uji MQTT telah direka untuk menilai kecekapan skema yang dicadangkan. Hasil kajian mendedahkan bahawa D-AES yang dicadangkan adalah lebih berpotensi daripada algoritma standard, dengan penambahbaikan sebanyak 8.75%, 10.45%, dan 6.81% dari segi keseimbangan, kesan longsoran, dan jarak hamming, masing-masing. Selain itu, keseluruhan skema menunjukkan sedikit peningkatan dalam overhed masa pelaksanaan sebanyak 0.99% dan 3.01% berbanding skema sedia ada untuk penerbit dan pelanggan, masing-masing. Walaupun skema yang dicadangkan meningkatkan overhed trafik sebanyak 90.57% untuk sepuluh atribut, terdapat tukar ganti yang jelas, iaitu penghantaran mesej MQTT yang selamat. Penambahbaikan yang dicadangkan secara berkesan meningkatkan kecekapan skema selamat berbanding kaedah sedia ada dan lebih sesuai untuk aplikasi secara praktikal dalam sistem IoT.

SECURE HYBRID SCHEME FOR SECURING MQTT PROTOCOL BASED ON ENHANCED SYMMETRIC ALGORITHM

ABSTRACT

Internet of Things (IoT) enables device and machine communication using TCP/IP protocol. Message Queuing Telemetry Transport (MQTT) is the most preferred protocol and is expected to be the *de facto* messaging IoT standard. Therefore, MQTT must achieve efficient security. Nevertheless, the most significant drawback of the MQTT is its lack of protection mechanisms which verifies only simple security objects such as non-encrypted authentication and authorization policies, and even there is no encryption mechanism. Data could be altered by intruders while in transit. Researchers have proposed various security techniques to address these issues. Meanwhile, the existing schemes for protecting the MQTT network have added processing overhead to the devices but remain vulnerable for various attacks. Therefore, this research work presented an integrated scheme known as "Secure Hybrid Scheme", to protect the MQTT protocol against any exploitations that might result in sophisticated cyberattacks. The proposed cryptosystem utilized two algorithms: a dynamic variant of the Advanced Encryption Standard (D-AES) and Key policy attribute base encryption (KP-ABE). A secure hybrid scheme introduces a new design architecture of the symmetric AES algorithm to encrypt the MQTT payload called "D-AES". The key expansion unit has been strengthened in the D-AES. Moreover, the values of the algorithm parameters are varied with each new key during the execution time of the algorithm. Thus, polymorphism has been achieved, while interoperability has been maintained. It has been demonstrated that the cipher is impenetrable to differential and linear cryptanalysis. Additionally, the

second part of the proposed hybrid cryptosystem is KP-ABE, which is utilized to cipher the private key of D-AES to avoid the computation overhead of bilinear maps. To undertake the experimentation, an MQTT test-bed was designed to evaluate the efficiency of the proposed scheme. The findings of this work revealed that the proposed D-AES is more promising than the standard algorithm, there exists an 8.75%, 10.45%, and 6.81% improvements in terms of balance, avalanche effect, and hamming distance, respectively. Besides, the overall scheme has a slightly increase in the execution time overhead by 0.99% and 3.01% from the existing scheme for publishers and subscribers, respectively. Although the proposed scheme increases the traffic overhead by 90.57% for ten attributes, there is an obvious tradeoff, which is secured MQTT message transmission. The proposed enhancements effectively improve the efficiency of the secure scheme compared to the existing works and are more suitable for practical application in IoT systems.

CHAPTER 1

INTRODUCTION

This chapter highlights the problem statements, research questions, objectives, and contributions in order to develop MQTT-based security solution for the IoT. The remaining sections of this chapter are organized as follows: The introduction is presented in Section 1.1. Section 1.2 explains IoT security. Section 1.3 presents the research motivation. The problem statement is stated in Section 1.4. The research goal and hypothesis are provided in Section 1.5. the research questions and objectives are described in Section 1.6. Section 1.7 includes the research contributions. Section 1.8 presents the research scope and limitations. Section 1.9 draws the research steps. Lastly, Thesis organization is stated in Section 1.10.

1.1 Introduction

This innovation of Machine-to-Machine (M2M) concepts or Internet of Things (IoT) is the predictable choice in the near future. The next revolutionary technology that is most likely to happen is transforming the traditional Internet into the future IoT that is fully integrated. According to Rose *et al.* (2015), IoT relates to enlarging the ability of non-computer devices such as sensors or actuators in terms of making mutual communications between these devices with no human intervention to generate, exchange and use data. Besides, IoT is uniquely identifiable things (sensors or actuators) connected through the network to control the status of the “Things” or retrieve its data (IEEE Initiativetive, 2015). The "Things" are programmable and provide services with an insignificant intervention of the human. Several definitions from IoT Corporation (IEEE Initiativetive, 2015; Rose *et al.*, 2014), explicate the major components of IoT which are the Internet and Things. The

Internet permits Things to communicate and by using standard communication protocols it gives humans the ability to communicate with Things. Accordingly, IoT, represented as Things, has a unique identifier that can be seamlessly blended within the environment and Internet connectivity makes it accessible by other objects (Things) to exchange data and achieve a common goal.

The IoT objects are expected to be on the rise, as stated by Ericsson (2022), where 24 billion IoT devices are interconnected and active in 2050. Besides that, the number of connected devices is speculated to exceed the number of human beings and its connectivity is increasing continuously. Examples of IoT objects (things or nodes) that have been used in a wide range of applications, to augment humans' conveniences are; smart sensors and health trackers (Islam *et al.*, 2015), smart home automation systems, and smart TVs (M. Miller, 2015), smart refrigerators (Dunn, 2017), IP cameras (Rajput., 2016), and others. Due to a large number of reasons, perpetrators can target IoT nodes. These IoT nodes are active on the Internet network (Heer *et al.*, 2011), remotely controlled (Li *et al.*, 2015), use insufficient security mechanisms due to resource limitations (Heer *et al.*, 2011), andz generate sensitive data that can be easily manipulated (Sadeghi *et al.*, 2015). Furthermore, due to reasons such as cost, time to ship, and easy user familiarity, security takes low priority by the vendors when planning to develop IoT devices (Penttinen, 2016; Rose *et al.*, 2014).

Due to the growing menace of cybercriminals, these cyber-attacks threaten IoT devices. A group of malware that targets smart nodes was reviewed in 2016 by a report from L3 Communications. The report indicates that bots that were hosted in Brazil, Colombia, and Taiwan; there are more than one million infected nodes in these bots (securityaffairs, 2016). Distributed denial of service (DDoS) attacks that were recently reported in many network infrastructures have been attributed to botnets that

are made up of infected IoT devices (FPAnalyst, 2016; Krebs, 2016). According to an article presented by (kerneronsec, 2016) explicated that more than 70 different vendors that provide Digital Video Recorder (DVR) services were exposed to run a remote code. Moreover, Cybersecurity Proofpoint in 2014 indicated that there were 100,000 consumer devices in a botnet including a refrigerator linked to the Internet, that dispatched 750,000 spam email messages to organizations and people all over the world (ProofPoint, 2014). A smart vehicle that was tested in 2014 was attacked by a buffer overflow and it was remotely controlled (Miller & Valasek, 2014). Therefore, according to the above-mentioned events, it is important to identify the threats that target the IoT nodes through the traditional Internet network and the need for a suitable security solution that should be properly deployed.

Typically, there are different types of hardware and software in the area of IoT; functionalities can rely upon the hardware, and managing them by software applications. Hence, IoT devices need to be able to easily communicate in order to manage all the smart nodes that are owned by the same user. Scientific literature has exhibited various IoT communication protocols to achieve bidirectional communication between devices and server/cloud (D2S) and between IoT devices (D2D) (Jaloudi, 2019). MQTT is the most widely adopted in IoT(Sanjuan *et al.*, 2020). MQTT uses minimal resources and consumes less energy (Nasser *et al.*, 2021). Exchanging messages among IoT nodes is facilitated by the MQTT Internet-facing broker/server. Hence, the security threats in the IoT domain that adopts MQTT needs to be identified to be protected.

In this thesis, the focus narrows on MQTT security issues in order to enable security features for the protocol. Besides, this research work enables MQTT as an application layer protocol with enhanced security features in order for encrypting the

protocol payload with the proposed secure hybrid scheme which has a polymorphic encryption feature for IoT devices considering the publish-subscribe communication paradigm. The most significant benefits of the proposed secure hybrid scheme are simplicity, performance efficiency, polymorphic feature, and robust security. The idea is relying on utilizing a less computational structure of the cryptosystem to be suitable for MQTT nodes. Unlike other techniques thus far that depend on traditional protocols of network security, the proposed scheme is intended to be practical from the perspective of the resource-constrained nature of IoT nodes.

1.2 IoT Security

It is a highly complicated job to secure IoT from a number of potential threats. However, when referenced under its layered architecture, it is manageable to some extent. Vulnerabilities and limitations exist in every layer that must be identified to guarantee the security of that layer by protecting it from various threats (Zhao & Ge, 2013). An appropriate and effective security system is required to address observed vulnerabilities of the IoT object and prevents such threats. In addition, malicious activities could be common on IoT networks because IoT devices may be reached from an untrusted network such as the Internet remotely. Consequently, sensitive data can also be revealed at any moment because of security vulnerabilities that haven't yet been fixed. Thus, security in the IoT sector must be taken into account and concerned, specifically for security goals as well as real deployments (Ahmad, 2016; Zhang & Li, 2014). It is important to zoom in for that on weakness and how it assists in a threat.

Vulnerability reflects the system's incompetence, which helps the intruder to detect the scope to breach system protection. This can lead to an increased risk which if ignored, can contribute to an attack. As stated by Top IoT Vulnerabilities (2016),

Table 1.1 lists the vulnerabilities list along with their contribution factors, which are potentially responsible for the occurrence of an attack on IoT machines. The results are for some of the vulnerabilities are caused through careless and irresponsible behavior in handling the IoT device. Therefore, there is a very important measure above all, that can easily be taken care and that is called self-awareness. In case of some unexpected behavior in the IoT nodes, the user should be well aware of all the hazards involved and there should be an appropriate intervention. To deal with smarter, newer attack-launch possibilities, the existing security system must be supplemented with features such as firewalls, content filtering, intrusion detection system, application whitelisting, and inspection technologies (Kolias *et al.*, 2017).

Table 1.1 IoT Arena Vulnerabilities and Their Responsible Weaknesses

Vulnerability	**Responsible weak points**
Lack of sufficient authentication and authorization	The intruder could have access to a certain interface with default weak passwords, weak password retrieval systems, lack of granular access control, and vulnerable protected credentials
Unreliable user interfaces	The controls or data can be reached by using no transport encryption, weak password retrieval systems, vulnerable of login credentials, and plain-text credentials
Insecure network services	Services of susceptible networks could be utilized to target a system or node by the attacker.
Privacy issues	The information which has weak protection could be reached by the intruder due to inadequate authentication, the service of the network not secure, lack of transport encryption, as well as unreliable interfaces.
Insufficient transport encryption/integrity verification	Inadequate transport encryption enables an attacker to access data sent across the network.
The inadequacy of the security configuration	The adversary can be able to reach control of the system and its data due to the lack of password options or encryption, and lack of permissions. Any node in the IoT network could be a cybercriminal.
Poor physical security	The adversary could access the operating system and data via memory cards, USB ports, and other storage/peripheral devices.

1.2.1 Identify IoT Security Goals and Security Attack

The most basic concepts in the IoT domain are clarified in this section: a security attack and a secure object (Mosenia & Jha, 2017). It is important to understand the security goal to distinguish the required security. Terms confidentiality, integrity, and availability are three main groups of traditional security and are known as the CIA triad in the state-of-the-art. Confidentiality is correlated with a series of rules in which the information can be accessed only by authorized entities. With the emergence of the IoT, the confidentiality of IoT-based MQTT objects is important because such objects can interact with sensitive information. For the service to be reliable in the IoT-based MQTT, the published data should be not modified and reached as it is by the subscriber and the subscriber has obtained only legitimate information as well as commands. Availability guarantees that the MQTT broker is accessible only by authorizable nodes in the MQTT network. Although the CIA triad is common, authors Cherdantseva and Hilton (2013) have shown that the CIA triad has failed to resolve recent threats that exist in a secure environment.

An IAS-octave, alluded to the Information Assurance and Security presented by the authors which are security goals with a comprehensive set, in terms of assurance and security investigated of a vast number of data. The security goal details are presented in Table 1.2 with their abbreviations and descriptions suggested by the IAS-octave. The security attacks and the secure object can be described as follows, once classified as major security goals.

- A secure object is an entity matched to or fulfilled all security goals.
- A security attack is an attack with at least one of the security targets being breached.

Table 1.2 IoT Security Requirements

Security goals	Description	Abbreviations
Confidentiality	Prevent the intruder from sniffing the data shared between the publisher and the subscriber and maintain privacy and data safety in general.	C
Integrity	Ensuring that the data transmission from a sender as it is and it should be not manipulated by the intruder node.	I
Nonrepudiation	An intermediate node could store the data packet and replay it later. Critical data may be used in the replay packet. Consequently, the detection of duplicate messages is required.	NR
Availability	The services of all IoT applications should be continually accessible and denied access by unauthorized systems or objects should not occur.	A
Privacy	The procedure in which policies or privacy rules are followed by an IoT system enables users to manage sensitive information.	P
Auditability	Guaranteeing the capability of the IoT system to do its activities with solid monitoring.	AU
Accountability	This procedure in which ensures that an IoT system makes the users responsible for their activities.	AC
Trustworthiness	Make sure that communicate with authentic nodes which are part of the network by confirming the other object's identities, so as to stop unauthorized access.	TW

1.2.2 IoT Data Protocols

This section paints a background of the common IoT data communication protocols. It should be noted that there is no data communication protocol adopted as a standard for IoT. Figure 1.1 depicts the chart of how Asynchronous Messaging Protocol (AMPP), Devices Profile for Web Services (DPWS), Constrained Application Protocol (COAP), and MQTT have enlarged these past fifteen years, which motivates the focus on MQTT.

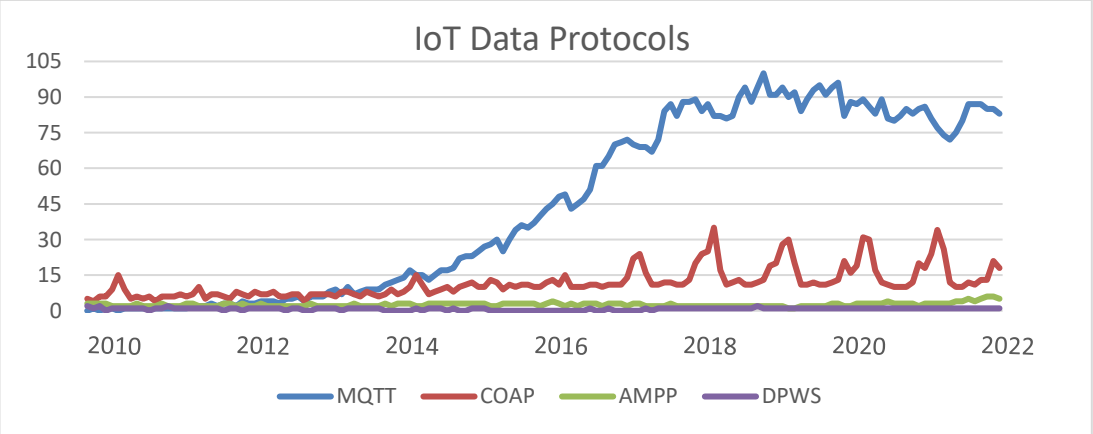

Figure 1.1 Google Trends of MQTT, COAP, DPWS, and AMPP (Google, 2022).

There are numerous comparisons of IoT data communication protocols, theoretically and feature-by-feature presented by the researcher's community as well as some of them focusing on performance evaluations and power efficiency in real implementations. The authors Karagiannis *et al.* (2015) provide a high-level study of IoT data communication protocols, but it does not include DPWS. Within the field of smartphone applications, the authors present a comparative study between MQTT and CoAP (De Caro *et al.*, 2013). Additionally, the authors Fysarakis *et al.*, (2016) have depicted a close study of MQTT, DPWS, and CoAP. The authors Thangavel *et al.* (2014) have shown the behavior of the protocols (CoAP and MQTT) when the conditions of the network are changed. Finally, three data communication protocols (OPC-UA, MQTT, and CoAP) were compared in the laboratory environment in the scenario of using cellular networks for communication (Dürkop *et al.*, 2015). Table 1.3 depicts the features comparison of all these protocols.

Table 1.3 Feature Comparison of The Main IoT Protocols

Features	**DPWS**	**XMPP**	**COAP**	**MQTT**
Version	1.1, OASIS (2009)	RFC 6120, IETF Saint-Andre (2011)	RFC7252,IETF Shelby *et al.*, (2014)	3.1.1,OASIS Standard (2014), ISO/IEC (2016)
Protocol Type	Service Oriented.	Message Oriented	Resource Oriented	Message Oriented
Transport	-TCP. -UDP.	TCP	-UDP -TCP planned	TCP
Synchronous	Yes (Service Invocation)	Near-real time	Yes (Request-response, via HTTP)	No
Asynchronous	-Publish-Subscribe to Service. -WS-Eventing.	Publish-Subscribe	Observe Resource - RFC 7641	Publish-Subscribe Topics
Discovery	WS-Discovery	XEP-0030	-RFC 5785 -RFC 6990	No
QoS	No	No	Limited	Three levels
Security	-Payload encryption. -WS-Security. -TLS/SSL. -IPSec. -802.15.4.	-SASL -TLS/SSL -Non-native End-to-End encryption	-Payload encryption. -DTLS. -IPSec. -802.15.4.	-Payload encryption. -TLS/SSL. - IPSec. -802.15.4.
Computer Resources	10Ks of RAM-Flash	10Ks of RAM-Flash	10Ks of RAM-Flash	10Ks of RAM-Flash
Power	Fair	Fair	Excellent	Good

The focus is on the MQTT protocol because it is considered as the de facto standard protocol that is deployed in various IoT systems; it is found in applications in several domains. Various libraries for several development platforms like Arduino are available in IoT, for different programming languages e.g., Python, Java, C, and Android and iOS for mobile platforms (Collina *et al.*, 2012).

From Table 1.3, it can be deduced that the main issue of DPWS is difficulty in implementing the dynamic discovery mechanism in a public network because of the limited range of UDP multicast messages to the local subnet. Another issue is due to multiple bidirectional data communication and data representation in eXtensible Markup Language (XML) format which generates massive traffic on the Internet and grows latency in application/device communications. However, some features of DPWS such as publish/subscribe eventing and dynamic discovery are limited in global device deployments due to many IoT gadgets. XMPP is not practical for M2M communications because it does not offer QoS options. However, XMPP messages follow XML structures and incur significant overheads to message sizes.

CoAP transactions can be protected only by the use of DTLS because of no built-in security options offered by the protocol. Since CoAP packet structure is diverse, hence, Hypertext Transfer Protocol (HTTP) servers might expose additional confusion when using DTLS. MQTT is designed to target limited resources devices and bandwidth limited communications and it is suitable for IoT and M2M to link devices/applications because MQTT employs various techniques for routing such as many-to-many, one-to-many, or one-to-one. MQTT is based on publish/subscribe programming model, runs on top of TCP/IP, consisting of three components which are subscribed, publish, and a broker. MQTT offers a quality of service (QoS) feature (D. Chen & Varshney, 2004) which is unique amongst IoT protocols. Moreover,

MQTT messages contain a mandatory fixed-length header (2 bytes) and an optional message-specific variable length header and message payload. It has limited security features in particular. Finally, according to Skerrett (2016), MQTT is considered as the second data communications protocol employed today by IoT.

1.3 Research Motivation

The Internet of Things, as a potential future innovation, is predicted to link billions of objects. However, the MQTT protocol in the IoT is a popular research topic and one such M2M data communication protocol, and it is also an OASIS standard. It has been developed to be lightweight for transporting messages in the publish-subscribe model suited for unreliable or high latency networks. The paradigm of Publish-Subscribe is commonly utilized as data communication for the IoT. Clients are more concerned with transmitting and retrieving data in this paradigm, and that they are less concerned with which exact end point is collecting the data.

In addition, simple authentication and authorization policies have been provided in the current implementations of MQTT, that could be utilized by the MQTT Subscriber node. The most significant weakness of the MQTT is because it has limited features of protection.

Furthermore, several approaches have been presented to improve the security of MQTT in the Internet of things, and numerous cryptosystems have been introduced to tackle the security issue of the MQTT; however, their usage is questionable because of hardware are not appropriate for such implementation of computational cost cryptosystems. In order to satisfy the need for protection while maintaining a minimum computational burden, a trade-off should be made. In light of this, an enhanced security solution to secure the MQTT there in IoT is urgently required.

1.4 Problem Statement

MQTT protocol is applied over the IoT arena and is considered as the de facto standard protocol that is deployed in various IoT systems; it is found in applications in several domains. Therefore, MQTT must achieve efficient security. The design of the MQTT protocol relies on a restful Application Programming Interface (API) which is hashed on HTTP and is prone to various cyber-attacks. Lack of security implementation in MQTT protocol. In addition, the underlying encryption algorithms for IoT are either computationally intensive (Frustaci *et al.*, 2017; Mades *et al.*, 2020) or vulnerable to a wide variety of statistical as well as other types of cryptanalysis attacks if the cryptosystem utilizes a weak key expansion algorithm (Ahamed *et al.*, 2019). Besides, if such key expansion produces subkeys with simple relationships, the entire cipher's secrecy will be compromised.

The current MQTT implementation verifies only simple security objects such as identity, authentication, and authorization policies (M. Singh *et al.*, 2015). However, the security of the MQTT protocol is based on a non-encryption authentication (Andy *et al.*, 2017; Anthraper & Kotak, 2019). Whether the authentication method is used by the broker or not, when data is transmitted between the broker and the MQTT node, the data still can be sniffed by the intruder. The traffic can be sniffed by the intruder via the publisher “Connect” packet node if the intruder with a publisher is at the same network. Thus, Information such as the public IP address of the MQTT broker, Port number, and payload data of the nodes are mostly targeted by the sniffing attacks. The username and password which is utilized to make a connection with a broker are existing in such a packet. In addition, “Keep Alive” packet can be tracked by the intruder. MQTT header is attached in this packet during

the process of the authentication which indicates how long the MQTT broker connection has remained with the IoT node.

Meanwhile, the existing approaches for protecting the MQTT network such as those proposed by Green *et al.* (2011), Ion (2013), Singh *et al.* (2015b; 2015a), Bisne and Parmar (2017a), and Katsikeas, *et al.* (2017) , and still vulnerable for various attacks such as differential and linear attacks, related-key attacks, SAT solver, algebraic attacks, Side-channel attack, meet-in-the-middle and SQUARE attacks. All these approaches employ the AES due to the promising performance in various platforms and could be adopted in the IoT network. It has been demonstrated that AES is vulnerable in a variety of IoT contexts (Rahman *et al.*, 2022).

Nevertheless, cryptanalysis can be adapted and run successfully because the encryption algorithm in its design nature has less complexity in the key expansion as well as the static nature of the SubBytes and ShiftRows transformation. This is due to the fact that the SubBytes transformation substitutes each byte in the state matrix with the value that is fixed in the substitution box. Moreover, the transformation of the ShiftRows shifts most of the bytes in each row of the state matrix cyclically to the left with a fixed offset. However, the first row of the state matrix is never shifted since the offset value of the first row is always zero. Therefore, there has been an increase in the variety of new threats.

Two strategies have suggested by Claude Shannon, confusion as well as diffusion for preventing cryptanalysis to employing statistical approaches (Shannon, 1949). According to Afzal *et al.* (2020), it has been demonstrated that the key scheduling algorithm of AES is lacking in the confusion and diffusion rate. In addition, the key expansion of AES includes minimal nonlinear features and a diffusion pattern that generates subkeys more slowly. The majority of cryptanalysis

attacks against AES exploit this flaw in its key expansion. Moreover, attacks such as differential and linear cryptanalysis as well as interpolation, related key, and square attacks have not been taken into consideration throughout the algorithm's construction. Since the standardization of AES in NIST FIPS-197 and thus its popularization to be utilized in a wide range of high-security applications, there has been extensive investigation into the vulnerabilities in the algorithm.

This has resulted in the development of extended types of threats and new kinds of the intruder, such as SAT solver and algebraic attacks, as well as hybrid and other attacks (Kaminsky *et al.*, 2010). The abovementioned problems have motivated researchers to propose approaches for securing MQTT. Therefore, a robust secure MQTT scheme that is aware of attacks must be proposed to secure MQTT messages in the network of IoT and protect the MQTT network from attacks.

1.5 Research Goal and Hypothesis

The main goal of this research study is to protect MQTT data in IoT, with reduced process overhead, energy consumption, and memory usage in constrained devices. Thus, by default MQTT can perform publish-subscribe without any security option or has limited security features such as only for simple authorization policies and basic authentication as well as preventing failures instigated by the intruder from occurring in the MQTT network. In order to attain the goal of this research, the following hypothesis has been stated as part of the research:

An integrated secure scheme for securing MQTT publisher-subscriber messages by preventing any exploitation of the MQTT messages by the cyber-attacks during the publishing-subscribing process between MQTT nodes in the IoT.

1.6 Research Questions and Objectives

Due to reasons as aforementioned in section 1.5, security implementation in MQTT protocol is still lack and prone to various cyber-attacks. As a result, three Research Questions (RQ) rise to address this research problem as follows:

1- What are the probable drawbacks of the currently available security mechanism?

2- What is an appropriate technique for securing MQTT messages during the process of publishing-subscribing in IoT?

3- Which suitable evaluation methods are suited for evaluating the intended scheme?

To fulfill the study's goals of responding to the above-mentioned research questions successfully, it is necessary to satisfy the Research Objectives (RO) as:

1- To enhance the key expansion of symmetric algorithm for improving the algorithm security level in terms of diffusion, confusion, and complexity with enough performance level to augment to the existing MQTT protocol.

2- To enhance the SubByte transformation of the AES algorithm to make it round key dependent.

3- To improve the shift row transformation of the AES algorithm to make it round key dependent and for improving the algorithm security level in terms of diffusion.

4- To propose a robust hybrid cryptosystem solution that integrates a security element with the current MQTT to strengthen its security.

1.7 Research Contributions

The MQTT protocol, which was recently standardized by OASIS, is such an M2M data communication protocol in the IoT arena (OASIS, 2009). It was developed to be relatively lightweight to be suitable to the sensors and mobile devices for message transport but many security elements have been violated. By default, there is no security option to secure the published data from the node like a sensor or actuator to the subscriber nodes. However, MQTT is vulnerable to attack because it does not utilize any protection scheme. Accordingly, a secure and integrated scheme is required for securing MQTT data in the IoT arena, the primary contribution of this work is to provide a secure and integrated scheme to improve the overall MQTT security where honesty among IoT objects is a critical issue.

The present work proposed a new mechanism called Secure Hybrid Scheme that is designed to address the security issues of the MQTT protocol in the Internet of Things. The detailed Research Contributions (RC) of the present work are as follows:

1- Strengthening the unit of key expansion to improve the strength of the symmetric algorithm at a very good scale through new advanced construction of the key expansion unit.

2- The symmetric algorithm's SubByte transformation is improved to be round key dependent, resulting in a key change that is easily detected in the cipher text.

3- Improvement of the symmetric algorithm's ShiftRow transformation in order to overcome the corresponding drawback and function dynamically and not rely on a static offset. As a result of the improved design, a better diffusion is achieved.

4- A new security option called a secure hybrid scheme is designed to provide confidentiality, access control, collusion resistance, and broadcast encryption to secure publisher-subscriber messages of MQTT-based IoT. In particular, the proposed secure hybrid scheme is able to provide at the same time confidentiality of payload for publisher and subscriber of MQTT-based IoT using the enhanced symmetric algorithm, fine-grained access control, collusion resistance, and broadcast encryption by using Attribute Based Encryption (ABE) algorithm, while not requiring publishers and subscribers to share keys.

1.8 Research Scope and Limitation

The scope of the proposed security scheme of this research is restricted for protecting an MQTT payload at the application layer, as demonstrated in Table 1.4 by protecting published data via encrypting the payload of the MQTT protocol to ensure Confidentiality in the MQTT network. The secure hybrid cryptosystem scheme is designed for securing MQTT devices that are employed in IoT network. However, an internal/external publisher and a subscriber can also exploit to cause various cyber-attacks on the MQTT broker. Since this research work is not focusing on the cyber-attacks that exhaust the resources of the MQTT broker. Therefore, cyber-attacks on the MQTT broker are not covered.

Table 1.4 Research Scope

Items		**Scope of Research**
Environment		IoT Devices
Target Layer		Application Layer
Attack Type		differential and linear cryptanalysis, Search Key, and Side-channel attack
Security Objects		Authentication, Confidentiality, and Privacy
Performance Metrics	Proposed algorithm	Balance, Hamming distance, avalanche effect, time
	Proposed scheme	Processing time, Storage overhead, Traffic overhead

1.9 Research Steps

Numerous research steps were being conducted in order to meet the objective of all this work, including the following: (i) Literature Review, to conduct a literature review and outline the research problem, (ii) Proposing a New Approach to secure the MQTT protocol in IoT, (iii) Design, Implementation the proposed approach, and (iv) Implementing and evaluating the finding; as shown in Figure 1.4 which depicts the research steps.

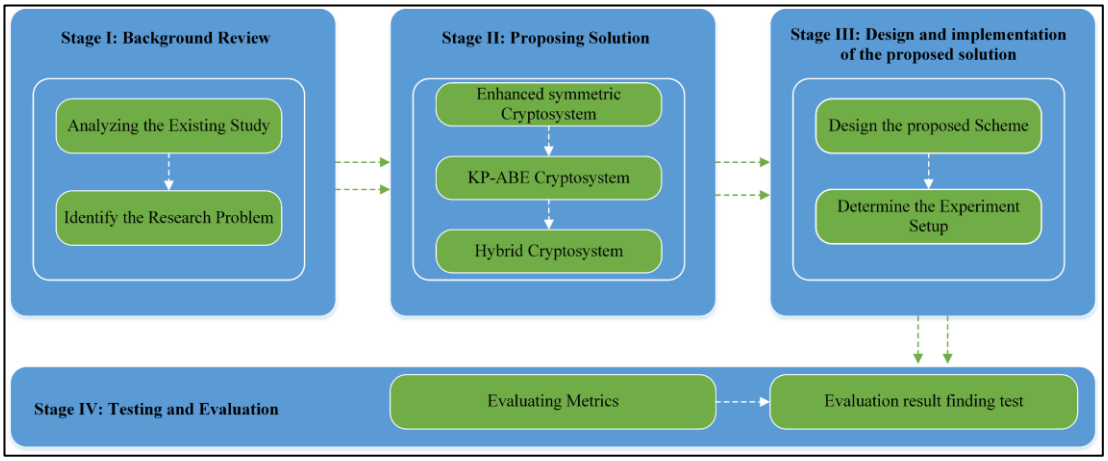

Figure 1.2 Major Stages of The Research Process

In the first stage, after conducting a comprehensive review of current techniques, the research problem is stated clearly and extensively analyzed. As a result, this literature review contributes to a better knowledge of the problem, the current solution area, and future research prospects.

In the second stage, a new design has been stated to solve the research problem of this thesis. This solution includes two components to protect MQTT nodes in IoT: An enhanced symmetric Algorithm and adopting the KP-ABE algorithm for making use of the properties of both algorithms. The proposed methodology and its implementation are depicted in the third stage.

The fourth stage is mainly concerned with testing and finding evaluations in order to accomplish the objectives of this research. The effectiveness of the proposed

work has been examined and evaluated in terms of processing time, storage usage, and traffic overhead, besides the security parameters of the enhanced symmetric algorithm I.e., balance, hamming distance, and avalanche effect. The secure hybrid scheme is compared with the existing mechanism to secure the MQTT protocol in the IoT arena.

1.10 Thesis Organization

The following is a rundown of each of the six chapters that make up this thesis.

Chapter 1: provides an overview of the IoT network and its related security issues, as well as an introduction to the IoT-based MQTT network. In this chapter, the research problem, objectives, and contribution are clearly presented.

Chapter 2: discusses the details and a comprehensive background of securing MQTT in the IoT. It also presents an insight majority of security solutions deployed for the MQTT protocol to offer a better understanding of the current study in the IoT.

Chapter 3: presents the design of the overall proposed hybrid scheme which includes the redesign of the AES algorithm to secure MQTT data in IoT. To illustrate the design and architectures at each phase, the steps of the methodology are described in sequential order and in detail.

Chapter 4: describes the design and the implementation of the proposed methodology for securing the MQTT protocol are presented in this chapter in detail. Focuses on the prototype model of the stages of the system as well as the implementation of the intended testbed.

Chapter 5: covers an in-depth analysis of the proposed D-AES algorithm as well as the overall performance of the proposed scheme based on the findings of the implementation work. The proposed methodology was evaluated. The achieved findings were compared to those existent methods.

Chapter 6: presents a conclusion regarding the research that was discussed in this thesis, as well as a summary of the entire discussion. This chapter also discusses a few prospective directions that are recommended for the future.

CHAPTER 2

LITERATURE REVIEW

This chapter includes details and a comprehensive background of securing Publish-Subscribe messages in IoT, particularly the MQTT protocol. It also presents an insight majority of security solutions deployed for the MQTT in order to offer a better understanding of the current study in the IoT. Additionally, this chapter presents a review are among the most widely used methodologies to secure MQTT in the IoT.

The remaining sections within this chapter are organized as: Section 2.1 presents the overview of the IoT. The underlying concept about MQTT protocol is clarified in-depth in Section 2.2. MQTT security is explained in detail in Section 2.3. The threat model of MQTT that are applicable to a standard MQTT-based IoT context are discussed in Section 2.4. MQTT attacks taxonomy and countermeasures are presented in detail in section 2.5. Section 2.6 describes the protection levels of the MQTT protocol for various security threats. Section 2.7 includes details mostly on related works for protecting the MQTT protocol. Section 2.8 presents the possible solutions. Similarly, the key features of the new security option for protecting MQTT data are stated in Section 2.9. Lastly, the chapter's summary is briefly described in Section 2.10.

2.1 Overview of IoT

In 1998, Kevin Ashton introduced the IoT concept (Borgohain *et al.,* 2015; H. Zhou, 2012). They declare that IoT are computers that are linked to each other over the Internet and knows everything about "things." IoT has the ability to collect and use data from these "things" without any guidance from a human being. In IoT, the electronic devices that are used frequently are interconnected to each other with the

capability of sensing and contextual knowledge (Hossain *et al.*, 2015; Singh & Singh, 2015). These devices are permitted to gather and give data. Besides that, domains such as smart environment, healthcare, transportation, industry, logistics as well as city information, social gaming robot, and personal lives have a wide range of IoT these days.

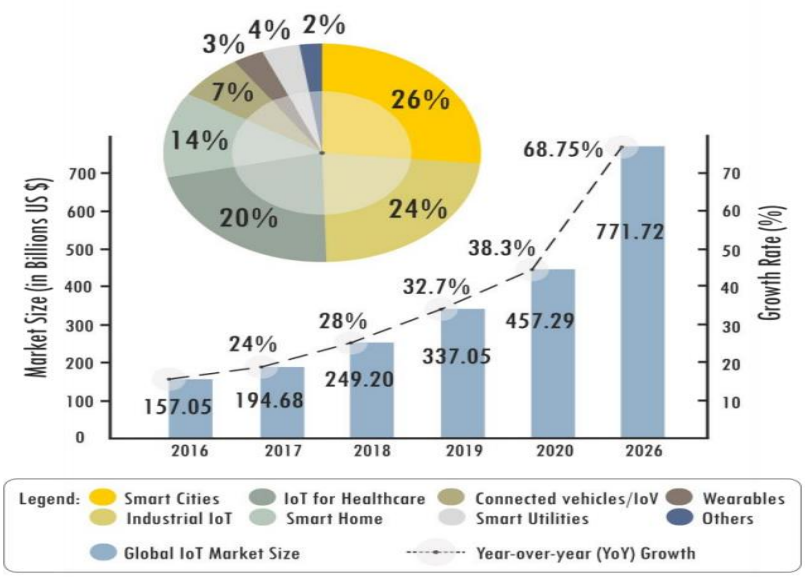

Figure 2.1 Global IoT Market Size 2016 - 2026 (Ray, 2018).

As depicted in Figure 2.1, the market of IoT has and will experience significant rapid growth as well as in the coming generations. The market of IoT value is expected to hit a market cap of 771 billion USD by 2026, starting from 157 billion USD in 2016 (GrowthEnabler, 2017; Stoyanova *et al.*, 2020).

2.1.1 IoT Infrastructure

The IoT platform includes a variety of intelligent objects that gather, process, compute, and interact with other intelligent objects. There are three layers in the IoT domain which are the application, network, and physical layer. There are also many things recently presented by industries that are embedded with smart things. As presented in Figure 2.2 some emerging technology is integrated with IoT. The IoT

technology could either be IoT-Fog-Cloud or IoT-Cloud based. For reliable IoT technologies, security concerns such as real-time monitoring (Casola et al., 2019), data privacy (Al-Hasnawi *et al.*, 2019), IoT test bed (Siboni et al., 2019), and machine-to-machine communication (Chen & Lien, 2014) should have to be discussed. The architecture of the IoT can be organized in a centralized, distributed, decentralized way. One of the most complicated issues in IoT applications is computing as well as processing in real-time. Cloud computing offers guaranteed data protection and more storage. Nowadays, most IoT application monitoring in real-time involves processing and computation at the network edge. This allows immediate action could be taken, such as pursuing serious fire detection, and the patient's health condition. It is much more prone to the adversary when processing and computation are done on the network edge utilizing fog devices, as these devices are limited resource devices, the conventional defense is not sufficient. A technique such as a machine learning has lately been adopted in IoT which makes it more intelligent and autonomous to make a decision.

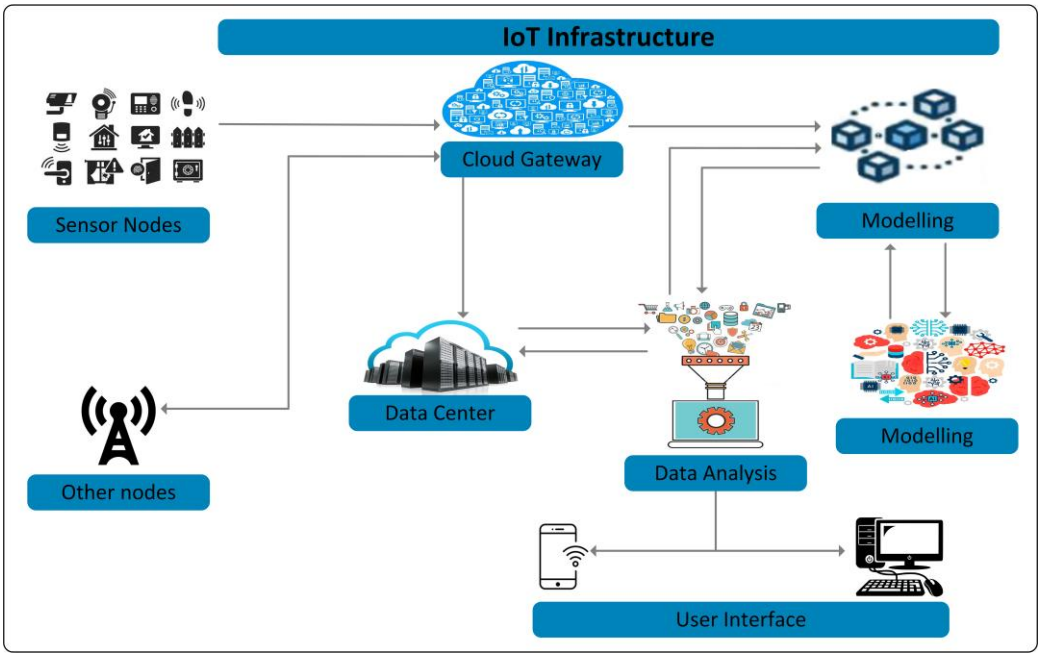

Figure 2.2 Internet of Things Infrastructure (Mohanta et al., 2020).

Various smart nodes are linked together to provide an application using certain common protocols such as the MQTT. To obtain the device tamper-proof and construct trust among end-users, IoT infrastructure security concerns need to be tackled. Utilizing an intelligent algorithm in the IoT platform could afford data interoperability (Nawaratne *et al.*, 2018).

2.2 Underlying Concept About The MQTT Protocol

MQTT is a lightweight data communication protocol based on the paradigm of publish-subscribe. This section discusses the protocol's fundamental concepts as well as its security-related features of the protocol.

2.2.1 Message Queuing Telemetry Protocol

MQTT is designed to target limited resources devices and bandwidth limited communications by IBM/Eurotech (IBM & Eurotech, 1999) and an OASIS standard (Standard, 2014). It has limited security features in particular; MQTT employs SSL/TLS to protect data communication and simple authentication (user-name/password) (HiveMQ, 2015). MQTT does not have discovery capabilities and is not asynchronous. Moreover, MQTT messages contain a mandatory fixed-length header (2 bytes) and an optional message-specific variable length header and message payload as illustrated in Figure 2.3.

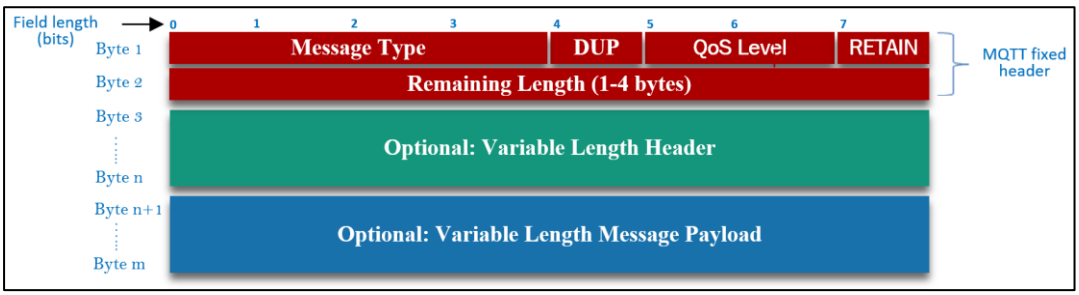

Figure 2.3 MQTT Header

MQTT is implementing a publish-subscribe programming model, as declared above; this implies that:

Topics: Classifies the information. Topic structures have no order or rules, but hierarchies are permitted via separators with the form of file system path called topic levels e.g., “office/light/status”.

Clients or nodes (getting access to the published data): Nodes can get access to act as reading action to the published date in a particular topic via subscribing to a similar topic. Clients should publish data under specific topics and can subscribe published data to other corresponding entities accessed to the same broker or server via that given topic. The procedure of subscription is achieved by relaying a SUBSCRIBE message command to the server or broker that is responsible to join the requesting node requiring access to that topic.

Nodes can generate content that acts as writing action: By publishing information to a specific topic, the published data procedure is achieved by relaying a PUBLISH message command to the broker that is not responsible for any treatment of the message transmitted. It sends the payload message to all nodes that earlier subscribed to the same topic. MQTT acts as one of the worldwide used protocols in different domains as shown in Table 2.1.

Table 2.1 Some Solutions That Use The MQTT Protocol

Smart home	**Brokers**	**Clients**
Homegear	Mosquitto	CocoaMQTT
Domoticz	ActiveMQ	emqttc
Lelylan	Hbmqtt	mqtt-client
cul2mqtt	HiveMQ	M2Mqtt
aqara-mqtt	Moquette	mqtt cpp
Home.Pi	Mosca	mqttex
Home Assistant	VerneMQ	Paho
Pimatic	Hrotti	rumqtt
FHEM	SurgeMQ	hbmqtt