# A Secure Integrated Framework for Fog-Enabled Cyber Physical Systems


**Aisha Kanwal Junejo**

Department of Computer Science,
School of Mathematics, Computer Science, and Engineering.
City University of London

This dissertation is submitted in partial fulfillment of the requirements for the degree of

*Doctor of Philosophy in Computer Science*

City University of London October 2019

To my parents, Shamsuddin Junejo and Shamim Junejo, without their prayers and support, this would not have been possible. I am indebted to my father who taught me that the best kind of knowledge to have is that which is learned for its own sake, and even the toughest task can be accomplished if it is done one step at a time. To my siblings, Nadir, Fareesa, Rabia, Khadijah, and Zain, for their emotional support during the course of this PhD.

# Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This dissertation is my own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Acknowledgements. This dissertation contains fewer than 65,000 words including appendices, bibliography, footnotes, tables and equations and has fewer than 150 figures.

Aisha Kanwal Junejo
October 2019

# Acknowledgements

All Praise to God for giving me the strength and will to complete this work.

I would like to offer my deepest gratitude to my first supervisor Dr Nikos Komninos who has been an ideal mentor. His guidance, motivation, insightful criticism, and patient encouragement aided the writing of this thesis in innumerable ways. I believe, his way of conducting my supervision has enabled me to become an independent researcher.

I am very thankful to my second supervisor Professor Lorenzo Strigini whose steadfast support of this work was greatly needed and deeply appreciated.

I am very grateful to the coordinators of Intact Erasmus Mundus project, Professor Bhawani Shankar Chowdhry (Mehran University of Engineering and Technology, Jamshoro, Pakistan), Professor Azizur Rahman (City University of London), and Professor Christos Themistos (Fredrick University, Cyprus) for awarding me the scholarship to conduct my research at the City University of London.

Finally, this PhD work would have not been possible without the support of my family and friends. I am immensely indebted to my beloved parents for allowing me to come to London and for being my strength emotionally. I am thankful to my landlady Dr. Faiqa Mazhar, Aunty Talat, Surjeet, and Nusrat for their love and emotional support that enabled me to fulfill my goal of completing this PhD. I will always appreciate all the presents Surjeet sent to me on festive occasions and never let me feel that I am alone in London.

# Abstract

The next generation of fog-enabled cyber physical systems (Fog-CPS) face numerous security, privacy and trust challenges. Establishing trustworthy and dependable Fog-CPS systems demand an integrated approach that adapts a multi-faceted and multi-dimensional solution strategy to countermeasure the challenges faced by the Fog-CPS systems. However, to the best of this researcher's knowledge, none of the existing studies had adopted an integrated approach to solve the challenges faced by the Fog-CPS systems. Considering the limitations of the existing studies, this research proposes an integrated framework for fog-CPS systems that addresses their security and trust challenges.

The proposed framework is comprised of two main components, 1) security component (SC) and 2) trust management system (TMS). The SC component guarantees that all entities of a Fog-CPS system i.e. fog nodes and cyber physical system (CPS) devices, have unique identities and only authorized parties can access the fog resources. The TMS component ensures that Fog-CPS entities are trustworthy. To be more specific, fog nodes are providing the acceptable quality of service based on the requirements of a specific Fog-CPS use case under consideration. Moreover, it also guarantees that CPS devices are not compromised and reporting actual communication parameters, namely, energy consumption, bandwidth and response time. The parameters reported by CPS devices are subsequently used as an evidence in trust computation for fog nodes.

As part of the SC, a novel lightweight encryption scheme based on elliptic curve cryptography is proposed to enforce robust authentication and authorization. The proposed scheme uses the inherent attributes of CPS devices to generate the cryptographic key pairs. The attributes belonging to CPS devices enables robust authentication and authorization. Unlike existing attribute based encryption (ABE) and identity based encryption (IBE) schemes, in the proposed scheme, each entity/CPS device generates its own public/secret key pair and does not need a certification authority (CA) to authenticate the public keys of other entities. Each CPS device can calculate each other's public keys, which are based on a shared attribute set.

Moreover, in the case of key revocation, the proposed scheme considers a light and efficient approach wherein the new keys are generated by incurring an overhead of only one

extra component. The experimental results of the proposed scheme demonstrate that it is computationally efficient compared to existing ABE schemes which are based on bilinear pairing and elliptic curves.

The TMS, the second component of the proposed framework, evaluates the performance of Fog-CPS entities based on a set of QoS parameters and network communication features. It subsequently computes their trust. Trust computation is formulated as a statistical regression problem, and the random forest regression is employed to solve it.

A Fog-CPS system is an inherently open and distributed, it is therefore vulnerable to collusion, self-promotion, bad-mouthing, ballot-stuffing and opportunistic service attacks. The compromised entities can impact the accuracy of trust computation model by increasing/decreasing the trust of other nodes. These challenges are addressed by designing a generic trust credibility model which can countermeasures the compromise of both CPS devices and fog nodes. The credibility of each newly computed trust value is evaluated and subsequently adjusted by correlating it with a standard deviation threshold. The standard deviation is quantified by computing the trust in two configurations of hostile environments and subsequently comparing it with the trust value in a legitimate/normal environment. Trust computation results demonstrate that credibility model successfully countermeasures the malicious behaviour of all Fog-CPS entities i.e. CPS devices and fog nodes.

The trust computed by the TMS component is incorporated in access control policies and ensures that only trusted entities are granted access to fog resources and collaborate with other entities in the system. The integration of two components, SC and TMS ensures that security and trust challenges of Fog-CPS systems are adequately addressed.

# Table of contents

**List of figures** **xvii**

**List of tables** **xix**

**Glossary of Abbreviations and Symbols** **xxi**

**1 Introduction** **1**

1.1 Research Background . . . . . 1

1.1.1 NIST Definition of CPS Trustworthiness . . . . . 5

1.1.2 Existing Approaches . . . . . 5

1.2 Research Motivation . . . . . 9

1.3 Overview of the Approach . . . . . 11

1.4 Research Question, Objectives and Hypotheses . . . . . 13

1.4.1 Research Question . . . . . 13

1.4.2 Research Objectives . . . . . 13

1.4.3 Research Hypotheses . . . . . 13

1.5 Contribution . . . . . 14

1.6 Thesis Outline . . . . . 14

1.7 Publications . . . . . 15

**2 Literature Review** **17**

2.1 Fog Computing in Cyber Physical Systems . . . . . 17

2.1.1 Cloud-Fog-Device Framework . . . . . 18

2.1.2 Features of Fog Computing . . . . . 20

2.1.3 Fog Computing Standards . . . . . 22

2.2 Integrated Frameworks . . . . . 23

2.3 Security Schemes . . . . . 27

2.3.1 Cryptographic Techniques . . . . . 27

2.3.2 Symmetric Key Encryption . . . . . 27

2.3.3 Asymmetric Key Encryption . . . . . 28
2.3.4 Cryptographic Hash Functions . . . . . 32
2.4 Elliptic Curve Cryptography . . . . . 33
2.4.1 Scalar Point Multiplication . . . . . 34
2.4.2 Finite Field Theory . . . . . 34
2.4.3 Elliptic Curve Cryptography Protocols . . . . . 36
2.5 Pairing Based Cryptography . . . . . 37
2.5.1 Bilinear Maps . . . . . 37
2.5.2 Properties of Pairing . . . . . 38
2.5.3 Decisional Bilinear Diffie-Hellman problem (DBHP) . . . . . 38
2.5.4 Security of Pairing-based Cryptography . . . . . 39
2.5.5 Pairing-Friendly Elliptic Curves . . . . . 39
2.5.6 Types of Pairing-Friendly Curves . . . . . 39
2.5.7 Common Ways to find Pairing-Friendly Curves . . . . . 40
2.5.8 Commonly used Pairings . . . . . 41
2.5.9 Choice of Pairing . . . . . 41
2.6 Pairing-based Encryption Schemes . . . . . 42
2.6.1 Functional Encryption Schemes . . . . . 42
2.6.2 Predicate Encryption with Public Index . . . . . 43
2.6.3 Attribute based Encryption Schemes . . . . . 43
2.6.4 Distributed Multi-Authority ABE Schemes . . . . . 44
2.6.5 Attribute Based Signature Scheme . . . . . 44
2.6.6 Predicate Encryption (PE) . . . . . 45
2.6.7 Anonymous Attribute Based Encryption . . . . . 45
2.6.8 Predicate Encryption Scheme . . . . . 45
2.6.9 Hidden Vector Encryption Scheme . . . . . 45
2.6.10 Functional Encryption Schemes Based on Elliptic Curves . . . . . 46
2.6.11 Discussion on Cryptographic Schemes . . . . . 46
2.7 Security Evaluation Techniques . . . . . 47
2.7.1 Provable Security . . . . . 47
2.7.2 Security Models . . . . . 48
2.7.3 Cryptanalysis Attack Models . . . . . 49
2.7.4 Security Notions for Cryptographic Schemes . . . . . 50
2.8 Related Studies in Cryptographic Techniques . . . . . 51
2.8.1 Related Studies in Cloud-Assisted Cyber-Physical Systems . . . . . 51
2.8.2 Related Studies in Constant-size ABE Schemes . . . . . 52

2.8.3 Related Studies in Elliptic Curve Based Encryption Schemes . . . . 53
2.8.4 Other Security Schemes . . . . . 54
2.9 Related Studies in Trust Models . . . . . 54
2.9.1 Classification of Trust Management Mechanisms . . . . . 57
2.9.2 Trust Models for Cloud Computing . . . . . 59
2.9.3 Trust Models for IoT Systems . . . . . 62
2.9.4 Trust Models for MANETs . . . . . 64
2.9.5 Trust Models for Cyber-Physical Systems . . . . . 66
2.9.6 Discussion on Trust Models . . . . . 67
2.10 Summary of the Chapter . . . . . 69

**3 A Secure Integrated Framework 71**
3.1 Fog-enabled Smart Power Grid Control System . . . . . 71
3.1.1 The Proposed Deployment Model . . . . . 72
3.1.2 Attacks on Cyber Physical Systems . . . . . 73
3.1.3 Security, Privacy and Trust Threats and Challenges in Fog-CPS Systems . . . . . 74
3.1.4 Security Properties Violation . . . . . 78
3.1.5 Key Challenges in Designing Secure and Robust Solutions for Fog-CPS Systems . . . . . 78
3.1.6 Security Requirements of Fog-CPS Systems . . . . . 79
3.1.7 Discussion . . . . . 81
3.2 The Proposed Secure Integrated Framework . . . . . 81
3.2.1 Security Component . . . . . 84
3.2.2 Trust Management System . . . . . 85
3.3 Proposed Security Component . . . . . 86
3.3.1 Set of Attributes . . . . . 87
3.3.2 Key Pair Generation . . . . . 87
3.3.3 Assumptions . . . . . 88
3.3.4 Preliminaries . . . . . 88
3.3.5 Fog-CPS Scheme Application in Fog-SGC Scenario . . . . . 91
3.3.6 Fog-CPS Description . . . . . 92
3.4 Mathematical Construction of Fog-CPS Scheme . . . . . 93
3.4.1 Partial Key Pair Generation$(\lambda, \mathbb{A}_S) \rightarrow PK_{\mathbb{A}_S}/SK_{\mathbb{A}_S}$ . . . . . 93
3.4.2 Final Public KeyGen$(PK_{\mathbb{A}_S}, \mathbb{A}_K) \rightarrow PK_{\mathbb{A}_K}$ . . . . . 94
3.4.3 Final Secret KeyGen$(SK_{\mathbb{A}_S}, \mathbb{A}_K) \rightarrow SK_{\mathbb{A}_K}$ . . . . . 95
3.4.4 Encrypt$(PK_{\mathbb{A}_K}, \mathbb{P}, M) \rightarrow CT$ . . . . . 95

3.4.5 Decrypt$(SK_{\mathbb{A}_K}, \mathbb{P}, CT) \rightarrow M$ . . . . . 96
3.4.6 Partial Key Pair Update$(\lambda, \acute{\mathbb{A}_S}) \rightarrow P\acute{K}_{\mathbb{A}_S}/S\acute{K}_{\mathbb{A}_S}$ . . . . . 99
3.4.7 Final Keys Update$(P\acute{K}_{\mathbb{A}_S}, \acute{\mathbb{A}_K}) \rightarrow P\acute{K}_{\mathbb{A}_K}/S\acute{K}_{\mathbb{A}_K}$ . . . . . 99
3.5 Theoretical Security Analysis and Evaluation . . . . . 101
3.5.1 Chosen Ciphertext Attack with in a Selective Security Game Model 102
3.5.2 Key Generation Analysis . . . . . 103
3.5.3 Network Devices Compromise Analysis . . . . . 106
3.6 Trust Management System . . . . . 107
3.6.1 What is Trust and Credibility in TMS? . . . . . 107
3.6.2 Trust Relationship . . . . . 108
3.6.3 Trust Notion . . . . . 108
3.6.4 Trust Computation . . . . . 108
3.6.5 Trust Distribution . . . . . 109
3.6.6 QoS Monitoring and Service Matching . . . . . 109
3.6.7 Objective Trust Computation . . . . . 110
3.6.8 Trust Parameter Monitoring . . . . . 112
3.6.9 CPS Device Parameter Monitoring . . . . . 112
3.6.10 Data Anomalies Detection . . . . . 112
3.6.11 CPS Device Trust Computation . . . . . 113
3.6.12 Trust Credibility Evaluation . . . . . 113
3.6.13 Fog Node Trust Computation . . . . . 115
3.6.14 Trust Computation for CPS Devices . . . . . 115
3.6.15 Trust Evidence Database . . . . . 116
3.7 Integration of SC and TMS . . . . . 116
3.8 Concluding Remarks . . . . . 117
3.9 Summary of the Chapter . . . . . 118

**4 Experimental Evaluation** **121**
4.1 Experimental Evaluation of SC . . . . . 122
4.1.1 System Configurations . . . . . 122
4.1.2 Implementation and Evaluation . . . . . 122
4.1.3 Memory Overhead . . . . . 134
4.1.4 Computational Overhead . . . . . 137
4.2 Experimental Evaluation of TMS . . . . . 138
4.2.1 Implementation Environment . . . . . 138
4.2.2 Dataset Generation . . . . . 138
4.2.3 Trust Label Generation . . . . . 139

4.2.4 Random Forest Regression Training and Testing . . . . . 140
4.2.5 Trust Results . . . . . 140
4.2.6 Fog-CPS Entities Trust Results . . . . . 140
4.2.7 Credibility Model Evaluation . . . . . 144
4.2.8 Resilience against Attacks . . . . . 151
4.2.9 Comparative Analysis . . . . . 152
4.2.10 Trust Computation Processing Time . . . . . 155
4.3 Concluding Remarks . . . . . 157
4.4 Summary of the Chapter . . . . . 159

**5 Conclusions and Future Work** **161**
5.1 Restating Research Problems and Research Goals . . . . . 161
5.2 Research Contributions and Distribution of Work . . . . . 162
5.2.1 A Secure Integrated Framework . . . . . 162
5.2.2 Fog-CPS Scheme . . . . . 163
5.2.3 Trust Management System . . . . . 163
5.3 Future Work . . . . . 164
5.4 Concluding Remarks . . . . . 165

**References** **167**

# List of figures


2.1 Three-Layer Architecture of Fog Computing (Ni et al., 2018) . . . . . 18
2.2 Classification Tree for Security and Privacy Technologies . . . . . 26
2.3 Symmetric Key Cryptography Based Communication System . . . . . 27
2.4 Public Key Cryptography Based Communication System . . . . . 28
2.5 Hierarchy of ECC Operations (Khan, 2015) . . . . . 34
2.6 Classification Tree of Trust Models . . . . . 56

3.1 Fog-Enabled Smart Power Grid Control . . . . . 73
3.2 Integrated Framework for Fog-CPS Systems . . . . . 83
3.3 Fog-CPS Scheme Application in Fog-SGC Scenario . . . . . 91

4.1 Experimental Set-up . . . . . 121
4.2 Final Key Pair Generation Fog-CPS Scheme (Algorithms 2 & 3), and Setup + KeyGen (other schemes) . . . . . 124
4.3 Partial Key Pair Generation (Algorithm 1) . . . . . 124
4.4 Final Key Pair Update (Algorithms 7 & 8) . . . . . 124
4.5 Partial Key Pair Update ( Algorithm 6) . . . . . 125
4.6 Processing time (sec) for Encryption (Message size = 1 KB) . . . . . 125
4.7 Processing time (sec) for Encryption (Message size = 1 MB) . . . . . 125
4.8 Processing time (sec) for Decryption (Message size = 1 KB) . . . . . 126
4.9 Decryption Algorithm (Message size = 1 MB) . . . . . 126
4.10 Final Key Pair Generation Fog-CPS Scheme (Algorithms 2, 3), and Setup + KeyGen (other schemes) . . . . . 127
4.11 Processing time (sec) for Partial Key Pair Generation Algorithm (1) . . . . 127
4.12 Processing time (sec) for Final Key Update Algorithms 7 & 8) . . . . . 128
4.13 Processing time (sec) for Partial Key Update Algorithm 6 . . . . . 128
4.14 Processing time (sec) for Encryption Algorithm 4 (1 KB Message) . . . . . 128
4.15 Processing time (sec) for Encryption Algorithm 4 (1 MB Message) . . . . . 129

4.16 Processing time (sec) for Decryption Algorithm 5 (1 KB Message) . . . . . 129
4.17 Processing time (sec) for Decryption Algorithm 5 (1 MB Message) . . . . . 130
4.18 Processing time (sec) of ECC-Auth-18 (Mahmood et al., 2018) Scheme . . 130
4.19 No: of Addition Operations . . . . . 131
4.20 No: of Multiplication Operations . . . . . 131
4.21 No: of Division Operations . . . . . 132
4.22 No: of Exponentiation Operations . . . . . 132
4.23 CPS Trust for Fog Nodes in Hostile Free Environment . . . . . 142
4.24 Objective Trust for Fog Nodes in Hostile free Environment . . . . . 143
4.25 Fog nodes Trust in Hostile Free Environment . . . . . 144
4.26 Fog Nodes High Trust without Credibility in Hostile Environment (***First*** Case $C_1$) . . . . . 146
4.27 Fog Nodes Low Trust without Credibility in Hostile Environment (***First*** Case $C_1$) . . . . . 146
4.28 Fog Nodes High Trust with Credibility in Hostile Environment (***First*** Case $C_1$) 147
4.29 Fog Nodes Low Trust with Credibility in Hostile Environment (***First*** Case $C_1$) 148
4.30 Fog Nodes High Trust without Credibility in Hostile Environment (***Second*** Case $C_2$) . . . . . 149
4.31 Fog Nodes Low Trust without Credibility in Hostile Environment (***Second*** Case $C_2$) . . . . . 150
4.32 Fog Nodes High Trust with Credibility in Hostile Environment (***Second*** Case $C_2$) . . . . . 150
4.33 Fog Nodes Low Trust with Credibility in Hostile Environment (***Second*** Case $C_2$) . . . . . 151
4.34 Comparative Analysis Normal Case . . . . . 153
4.35 Comparative Analysis Case-2 . . . . . 154
4.36 Comparative Analysis Case-3 . . . . . 155
4.37 Trust Computation Overhead . . . . . 156
4.38 Trust Computation Overhead . . . . . 157

# List of tables


2.1 Key Sizes for Symmetric and Asymmetric Cryptosystems (bits) (Lenstra and Verheul, 2001) . . . . . 34
2.2 Comparison of Pairings . . . . . 42

3.1 Attributes Shared between CPS Devices and Fog Assist Node . . . . . 87
3.2 Fog Node Trust Parameters . . . . . 112
3.3 Attributes of Fog-CPS entities and Trust Integration . . . . . 116

4.1 Memory Overhead . . . . . 133
4.2 Computational Overhead . . . . . 136
4.3 Simulation Parameters . . . . . 137
4.4 Trust Results . . . . . 141
4.5 Access Control Rights based on Fog Node Trust . . . . . 143
4.6 Collusion Attack Scenarios . . . . . 144
4.7 Trust Credibility and Access Rights (High Trust) . . . . . 148
4.8 Trust Credibility and Access Rights (Low Trust) . . . . . 149

# Glossary of Abbreviations and Symbols

## List of Abbreviations

**ABE:** Attribute Based Encryption

**ABS:** Attribute based Signature

**ACK:** Acknowledgement

**ADL:** Activities of Daily Living

**AEMS:** Advanced Electricity Monitoring Systems

**AES:** Advanced Encryption Standard

**AHP:** Analytic Hierarchy Process

**BAN:** Body Area Networks

**BDHP:** Bilinear Diffie-Hellman Problem

**BI:** Business Intelligence

**CA:** Certification Authority

**CCA2:** Adaptive Chosen-ciphertext Attack

**CCA:** Chosen-ciphertext Attack

**CDHP:** Computational Diffie-Hellman Problem

**CM:** Complex Multiplication

**COA:** Ciphertext-only Attack

**CP-ABE:** Ciphertext Policy Attribute Based Encryption

**CPA:** Chosen-plaintext Attack

**CPS:** Cyber Physical System

**CTMC:** Continuous Time Markov Chains

**DBHP:** Decisional Bilinear Diffie-Hellman Problem

**DDHP:** Decisional Diffie-Hellman Problem

**DDoS:** Distributed Denial of Service

**DIDS:** Distributed Intrusion Detection System

**DLP:** Discrete Logarithm Problem

**DNS:** Domain Name System

**DoS:** Denial of Service

**ECC:** Elliptic Curve Cryptography

**FA:** Fog Assist

**FE:** Functional Encryption

**Fog-CPS:** Fog-enabled Cyber Physical System

**Fog-SGC:** Fog-enabled Smart Power Grid Control System

**FSAW:** Fuzzy Simple Additive Weighting

**GDPR:** EU General Data Protection Regulation

**GNE:** Gestalt Nash Equilibrium

**HBFFOA:** Hypergraph Binary Fruit Fly Optimization Based Service Ranking Algorithm

**HIDS:** Host-based Intrusion Detection System

**HVE:** Hidden Vector Encryption

**IAM:** Identity and Access Management

**IBC:** Identity Based Cryptography

**IBE:** Identity Based Encryption

**INCAS:** Information Centric Adaptive System

**IND-CCA2:** Indistinguishability under Adaptive Chosen Ciphertext Attack

**IND-CCA:** Indistinguishability under (non-adaptive) Chosen Ciphertext Attack

**IND-CPA:** Indistinguishability under Chosen Plaintext Attack

**IoT:** Internet of Things

**IP:** Internet Protocol

**KGA:** Key Generation Authority

**KP-ABE:** Key Policy Attribute Based Encryption

**KPA:** Known-plaintext Attack

**M2M:** Machine to Machine

**MAC:** Media Access Control

**MADM:** Multi-attribute Decision Making

**MANET:** Mobile ad hoc Network

**MLR:** Multiple Linear Regression

**MPC:** Secure Multi-party Computation

**MSE:** Mean Square Error

**NAN:** Neighborhood Area Network

**NIDS:** Network-based Intrusion Detection System

**NIST:** National Institute of Standards and Technology

**OBDD:** Ordered Binary Decision Diagram

**ODE:** Ordinary Differential Equation

**OFC:** Open Fog Consortium

**OSF:** Open Semantic Framework

**P2P:** Peer to Peer Network

**PBC:** Pairing based Cryptography

**PE:** Predicate Encryption

**PEPA:** Performance Evaluation Process Algebra

**PET:** Privacy Enhancing Techniques

**PKE:** Public Key Encryption

**PKG:** Private Key Generator

**PPCG:** Privacy-preserving Computational Geometry

**PPDM:** Privacy-preserving Data Mining

**PPDQ:** Privacy-preserving Database Query

**PPID:** Privacy-preserving Intrusion Detection

**PPSC:** Privacy-preserving Scientific Computation

**QoS:** Quality of Service

**RA:** Reference Architecture

**RFID:** Radio-Frequency Identification

**RFR:** Random Forest Regression

**RO:** Random Oracle

**RSA:** Rivest, Shamir, and Adelman

**RTU:** Remote Terminal Units

**SC:** Security Component

**SCADA:** Supervisory Control and Data Acquisition

**SDN:** Software-defined Networks

**SDx:** Software-defined Anything

**SLA:** Service Level Agreement

**SP:** Service Provider

**SR:** Service Requester

**SSL:** Socket Layer Security

**TA:** Trusted Authority

**TESM:** Trust Enhanced Secure Model

**TIA:** Trust Information Agent

**TLS:** Transport Layer Security

**TMS:** Trust Management System

**TPM:** Trusted Platform Module

**VANET:** Vehicular ad hoc Network

**WAN:** Wide Area Network

**WSN:** Wireless Sensor Networks

# List of Symbols

$\acute{c}_i$ ith coefficient in polynomial of decryption algorithm

$\lambda_{param_i}$ Transparency of $param_i$ parameter

$\acute{\sigma}$ Standard deviation of $T$ over time duration $\tau$

$\acute{\sigma}$ standard deviation of $T$ over a time duration $\tau$

$\alpha, r_u, t_u$ Secret numbers

$\ddot{p}$ Pairing operation

$\gamma$ weight factor for $T^{fa \to fog}$

$\lambda$ Security parameter

$\mathbb{A}$ Device attribute set

$\mathbb{A}_K$ Shared attribute set

$\mathbb{A}_S$ Secret attribute set

$\mathbb{D}$ Trust Domain

$\mathbb{G}_{\mathbb{T}}$ Target Group

$\mathbb{G}$ Elliptic curve group generated by $P$

$\mathbb{G}_1, \mathbb{G}_2$ Additive Cyclic Groups

$\mathbb{P}$ Access Policy, $\mathbb{P} \subseteq \mathbb{A}_K$

$\mathbb{S}_K$ Attribute String for $\mathbb{A}_K$

$\mathbb{S}_P$ Attribute String for $\mathbb{P}$

$\mathscr{O}_{p,a}$ Key generation oracle

$T^{cps \to fog}$ Fog node trust based on CPS device parameters

$T^{fa \to fog}$ Fog node objective trust computed by FA

$T^{fog \to cps}$ CPS device trust based on fog node parameters

$T^{fog}$ Fog node trust

$T_t$ Trust at time instance $t$

$\perp$ Random element

$\phi$ SLA Contract

$\tau$ Time duration for trust evaluation

$c(i, fog)$ parameter report sent by ith CPS device for fog node

$c_i$ ith coefficient in polynomial of encryption algorithm

$cps$ CPS device

$CT$ Cipher-text

$e$ Bilinear bilinear mapping

$E_p(a,b)$ An elliptic curve $y^2 = x^3 + ax + b \ (mod p)$ defined over $Z_p$

$E_{F_q}$ Elliptic Curve over finite field $F_q$

$exp$ Exponentiation operation

$F_q$ Finite field of order q

$fog$ fog node

$G$ Cyclic Group

$g$ Group generator

$GF$ Galois field

$H_1(), H_2(), H_3(), H_4()$ Four one-way collision resistance hash functions

$k$ Embedding degree of $F_q$

$K_A$ Secret Key of A

$K_B$ Secret Key of B

$K_d$ Decryption Key

$K_e$ Encryption Key

$KDF()$ Key derivation function

$M$ Message

$P+Q$ Elliptic curve point addition

$P$ A base point in $E_p(a,b)$

$p$ A sufficiently large prime number

$P_i, U_i$ Public key components

$param_i$ $i$th Parameter in SLA $\phi$

$PK_{\mathbb{A}_K}/SK_{\mathbb{A}_K}$ Final Public and Secret keys generated on $\mathbb{A}_K$

$PK_{\mathbb{A}_S}/SK_{\mathbb{A}_S}$ Partial Public and Secret key pair generated on $\mathbb{A}_S$

$t$ Time stamp for QoS evidence gathering

$t_i$ ith time instance

$u_1$ Secret key components

$xP$ Scalar multiplication, $P \in E_p(a,b)$

$Z_p$ Finite field $Z_p = \{0, 1, ..., p-1\}$

$Z_p^*$ Finite field $Z_p = \{1, ..., p-1\}$

h Cofactor domain parameter of $E_{F_q}$

# Chapter 1

# Introduction

## 1.1 Research Background

Considering recent research and development endeavours towards urbanization, future cyber physical systems (CPS) are expected to be much more complex particularly in terms of scalability, heterogeneity of security requirements and resource availability. Moreover, it is estimated that in the near future, CPS systems will scale exponentially. According to Cisco, 50 billion devices will be connected to the Internet by 2020 (Evans, 2011) and this number will reach 500 billion by 2025 (Camhi, 2015). Besides that, by 2021, the data produced by interconnected devices will reach 500 zettabytes (ZB). However, the global data centres have the capacity to handle only 20.6 ZB of IP traffic (Cisco, 2018).

The above statistics suggest that the current cloud-based architectural approaches cannot sustain the projected data velocity and volume requirements of future CPS systems. Precisely, unfettered cloud-only solutions are impractical for many real-world CPS use cases, namely smart grid, smart traffic and smart cities, which generate a massive amount of sensory and contextual data. Considering the forecast exponential growth, the CPS deployment model has shifted from cloud to fog.

Fog computing and the CPS paradigm complement each other and are the enabler of next generation CPS systems. The next generation fog-enabled cyber physical systems (Fog-CPS) extend the functionality, features and capabilities of fog computing and CPS systems. Fog-enabled solutions are more suitable for CPS systems as the services with reduced latency, network jitter and response time are located near the edge of the network (OpenFog Consortium Architecture Working Group, 2017).

Despite the opportunities provided by Fog-CPS systems, they face increased security, privacy and trust challenges. CPS devices are vulnerable to cybersecurity challenges and threats due to their resource limited capabilities and lack of protection mechanisms. In Fog-

CPS systems, inter-device and inter-system collaborations are subject to risk and uncertainty. The compromise of one component can trigger malicious behaviour in other interacting components.

Recent cyber attacks on CPS systems including the Ukrainian power grid (Lee et al., 2016), DNS provider Dyn (Lewis, 2017) and the Stuxnet worm (Kushner, 2013) attack on Iranian nuclear facilities underline the threat of connectivity and vulnerability of resource constrained devices to be compromised. For instance, in the case of Dyn, the distributed denial of service attack was launched by small compromised devices which formed a botnet and collapsed the fundamental infrastructure comprising the Internet. In Stuxnet (Kushner, 2013), the worm stealthily spread between computers running Windows by exploiting the privilege escalation vulnerabilities.

Moreover, recent breaches in cloud also indicate the security and privacy challenges that Fog-CPS systems might face. The year 2014 in particular witnessed quite a number of security attacks. In Sage security breach (BBC News, 2016), personal information of employees working at 280 organizations was compromised. The security breach occurred as a result of "unauthorised access" by someone using an "internal" company login. In November 2014, the network of Sony corporation (BBC News, 2014) was attacked by a malware. Hackers accessed data including personal information of employees, financial details, email accounts and unreleased films. Additionally, hackers destroyed thousands of Sony's computers and hundreds of its servers following the attack.

The Home Depot (Krebs, 2014) security breach was another high level cybersecuity incident in which 56 million credit and/or debit cards and 53 million email accounts were compromised. Hackers also gained access to Home Depot's systems for nearly a six month period from April to September 2014. Moreover, in the first iCloud (Lewis, 2014) attack, hackers broke into the personal Apple accounts of celebrities and publicized their private photographs on the web. Images were believed to had been obtained via a breach of Apple's cloud services suite iCloud. The lack of two-factor authentication was deemed to be the reason behind unauthorized access. The Target security breach compromised up to 70 million customers' credit card information during the holiday season of 2013. The hackers gained access by stealing credentials from a third-party HVAC company through phishing emails. This network breach revealed many holes in the security strategy of the company.

As the communication in Fog-CPS systems is analogous to the communication between CPS devices and cloud, similar attacks could also be launched in Fog-CPS systems. The above mentioned attacks underline that Fog-CPS systems could become new targets for hackers and could bring enormous risks to their availability and reliability. More specifically, in Fog-CPS systems, machine-to-machine (M2M) communication takes place between

autonomous embedded devices (i.e. sensors and actuators) or between sensing devices, fog nodes and cloud services. M2M communication can be compromised via several attacks such as *interception* (i.e. eavesdropping), *interruption* (i.e. DoS attacks), *modification* (i.e. packet payload manipulation in transit) and *fabrication* (i.e. man-in-the-middle attack). The above attack mechanisms are results of the lack of ***confidentiality***, ***authentication*** and ***authorization***. Therefore, it is essential to secure the communication such that no malicious entities can purposefully manipulate it.

Moreover, a few of the other challenges have their roots in the unique features of fog computing such as decentralized infrastructure, mobility support, location awareness and low latency. To be more specific, in Fog-CPS systems, the fog nodes provide cloud services including, compute, storage, network and security to CPS devices and other components. As the fog nodes are deployed near the edge of the network, the end devices provision services from nearby fog nodes. Resultantly, the sensitive information of users ends up being shared, processed and stored on fog nodes which subsequently leads to the leakage of identity, data, usage and location information (Ni et al., 2018). These various details can be correlated to generate user profiles and further risk the privacy of individuals.

Furthermore, location based services are prevalent in Fog-CPS systems, they can leak the trajectory information of users and/or devices. Sharing of location information is inevitable in such systems and therefore demand sophisticated privacy preserving solutions which maintain a balance between user privacy and data utilization. The above mentioned facts highlight that privacy is a critical issue in fog-CPS systems and is way more complex than cloud computing and CPS systems operating in a controlled environment.

Likewise, the decentralized architecture and mobility support features of fog computing introduce trust issues in Fog-CPS systems. Unlike cloud servers, there is no predefined network architecture for Fog-CPS systems, the fog nodes can join and leave the network depending upon resource requirements of a specific use case under consideration. Some fog nodes which were part of the network could no longer exist and need to be replaced with other fog nodes. This could happen for various reasons including inability to provide quality of service (QoS), load balancing, service cost, energy usage, node capture or some other attacks. Additionally, similar to CPS devices, fog nodes are also deployed in open and untrusted environments and are therefore at risk of being compromised, broken and stolen.

Precisely, before any prior interaction, both CPS devices and fog nodes have no idea about how their partners will behave. Without trust mechanisms, such interactions are subject to risk and uncertainty that an entity might experience. The unintended outcomes of Fog-CPS systems can be subject to environmental disruption, operation in untrusted locations, human user and operator error, and attacks by hostile parties.

Having discussed the various aspects related to Fog-CPS systems, it is clear that establishing trustworthy systems is not trivial. Such complex scenarios require solutions that guarantee the Fog-CPS systems do not deviate from their designed behaviour when operating in hostile environments or in events of compromise. This researcher believes a secure integrated framework that takes into consideration the security and trust characteristics can address the challenges faced by the Fog-CPS systems. It can also alleviate the risk and uncertainty associated with interactions in open and distributed Fog-CPS environments.

The Fog-CPS systems face numerous challenges (mentioned above) which could be addressed by employing novel security mechanisms. However, this research focuses on addressing only *interception*, *modification* and *fabrication* attacks. Protection against interruption (DoS) attacks is out of the scope of this research. It is believed that the security goals i.e. ***data confidentiality***, ***authentication***, and ***authorization*** can guarantee protection against *interception*, *modification* and *fabrication* attacks. Precisely, a new attribute based encryption scheme is proposed to achieve ***data confidentiality***, ***authentication***, and ***authorization*** in a fog-based system. Precisely, in Fog-CPS systems, ***data confidentiality*** measures are required to ensure that only authorized fog nodes and CPS systems have access to the sensitive data. ***Authentication*** mechanisms establish the identities of fog nodes and CPS systems, and prevent the fake entities from joining the network. Likewise, ***authorization*** mechanisms specifically access policies ensure no unauthorized party can access the fog resources.

Additionally, the dependable behaviour of the Fog-CPS systems can be guaranteed by robust behavioural monitoring. Ensuring predictable system behaviour based on a set of predetermined conditions is very challenging. However, the interactions of the collaborating CPS devices and fog nodes can be evaluated based on various dimensions such as ***service quality***, ***competence***, ***integrity***, ***benevolence***, ***honesty***, and ***capability***. Each of the above mentioned dimensions of trustworthiness is an indicator of the system behaviour and guarantees that CPS systems and other components are performing as expected. After monitoring the interactions, trust can subsequently be computed to evaluate the performance and/or behaviour of a Fog-CPS system under consideration.

From the above discussion, it is clear that multitude of problems faced by the Fog-CPS systems require a multi-dimensional and integrated solution strategy. Proposing a solution which focuses on a single aspect, i.e. security, trust, safety, or reliability is not desirable. This researcher believes achieving the security goals and the trustworthiness properties would make certain that Fog-CPS entities are invulnerable to malicious attacks and environmental disruptions. Such an approach would also ensure the dependable performance of fog nodes. To be more specific, data and processes running on fog nodes could be completely decoupled and could not be modified and/or misused by malicious parties in the events of node mobility.

### 1.1.1 NIST Definition of CPS Trustworthiness

Adapting an integrated approach is also aligned with the recommendations of the NIST CPS framework. NIST, in June 2017, published a report with title "Framework for Cyber Physical Systems: Volume 2, Working Group Reports" (NIST, 2017). This publication covers different aspects related to cyber physical systems (CPS), 1) vocabulary and reference architecture, 2) cybersecurity and privacy, 3) data interoperability, 4) timing and synchronization, and 5) use cases. The report of the cybersecurity and privacy sub-group highlights the ***security*** and ***privacy*** challenges arising due to the interaction of heterogeneous and distributed components and systems in the cyber and physical worlds. It is further maintained that the notion of chain of trust is essential for CPS environments that contain diverse hardware and software systems and need to preserve integrity to perform mission-critical tasks. Moreover, the NIST report defines the trustworthiness of CPS systems as follows:

***"Trustworthiness is the demonstrable likelihood that the system performs according to designed behaviour under any set of conditions as evidenced by characteristics including, but not limited to, safety, security, privacy, reliability and resilience."*** (NIST, 2017)

However, each of these trustworthiness properties could itself be a research topic and their integration could be another research work. Considering the wide scope of CPS systems trustworthiness, this work focuses on security and trust and demonstrate how their integration can lead to a trustworthy and dependable Fog-CPS system. The purpose of discussing the NIST CPS framework is to highlight that this research work is timely and aligned with their recommendations.

### 1.1.2 Existing Approaches

Next, existing approaches to address the security and trust challenges of the Fog-CPS systems are discussed. To the best of this researcher's knowledge, no secure integrated framework exists for Fog-CPS systems in the literature. A few recent studies have proposed some frameworks for IoT, Cloud, WSN and VANETs etc. As the Fog-CPS systems also share many commonalities with the above mentioned systems the frameworks proposed for these systems have also been discussed. Moreover, related security schemes and trust models which are proposed for Fog-CPS systems are also discussed.

### Integrated Frameworks

Hao et. al (Wu and Wang, 2018) propose a collaborative game theory based security detection framework for IoT. The framework models the confrontation between an attacker and defender to identify the security events. Moreover, collaborative information sharing is achieved by the consensus protocol. George et. al (George and M. Thampi, 2018) adopt a graph modelling approach to identify the relationship between vulnerabilities in an industrial IoT system. The security issues are formulated as graph-theoretic problems. The study also proposes some risk mitigation strategies which can detect and remove the attack paths with high risk and low hop-length. Muhammad et. al (Muhammad et al., 2018) propose a surveillance framework for IoT systems using probabilistic image encryption. The study uses a video stigmatization method to extract the information frames and further encrypts the images to prevent data modification attacks.

Mick et. al (Mick et al., 2018) propose a lightweight authentication and secure routing scheme for smart cities. The study advocates that named data networking is more suitable for IoT than a host-centric Internet Protocol (IP) model. Cha et. al (Cha et al., 2018) propose a privacy framework in which users can set their privacy preference on Bluetooth low energy based applications. Fadi et. al (Al-Turjman et al., 2017) propose a key agreement framework which is based on the mobile-sink strategy and extends user authentication to the cloud-based applications. The proposed framework is based on bilinear pairing and elliptic curve cryptography (ECC). Chen et al. (Chen et al., 2015b) propose a distributed authentication framework for the multi-domain M2M environment. The proposed framework employs a hybrid encryption scheme involving identity based encryption (IBE) and advanced encryption standard (AES).

Alhanahnah et. al (Alhanahnah et al., 2018) propose a context-aware multifaceted trust framework for evaluating the trustworthiness of cloud services. The overall trust in a cloud service is computed by considering both service characteristics and user perspective. Service level agreement (SLA) based trust is computed by employing analytic hierarchy process (AHP) whereas the non-SLA based trust is computed by fuzzy simple additive weighting (FSAW). Namal et. al (Namal et al., 2015) propose a trust management system (TMS) for cloud-based IoT applications. The study employs "Weighted Sum" for trust aggregation and considers multi-dimensional parameters, namely availability, reliability, capability and response time for trust formation.

### Security Schemes and Trust Models

Next, some related security schemes and trust computation models are discussed. To secure the communication in cloud-assisted CPS systems, the literature adopts a hybrid approach in which both symmetric (AES-based) and asymmetric (RSA-based and ABE) encryption techniques are used. AES (Advanced Encryption Standard) is used to encrypt the communication between sensor nodes and the gateway, whereas the asymmetric schemes are used to encrypt the communication between the gateway and the service provider.

Recently, Mahmood et al. (Mahmood et al., 2018) proposed an authentication scheme for smart grid. The major limitation of the proposed scheme is the scalability in case of large-scale CPS systems. Kocabas et al. (Kocabas et al., 2016) present a medical cloud-assisted CPS architecture consisting of acquisition, pre-processing, cloud, and action layers. The study proposes an AES (Advanced Encryption Standard) symmetric key encryption scheme for communication between acquisition and pre-processing layers. The key disadvantage of (Kocabas et al., 2016) is the key management of symmetric keys in such a complex environment. Yeh et. al (Yeh et al., 2018) propose a variant of a ciphertext-policy attribute based encryption (CP-ABE) scheme for eHealth systems. Similar to existing ABE schemes, the proposed CP-ABE scheme employs fine-grained access control in cloud-based personal health care applications. The major limitation of the proposed scheme is the scalability in case of large-scale IoT systems.

Sravani et. al (Challa et al., 2017) present a signature-based authenticated key establishment scheme for IoT applications. Chunqiang et. al (Hu et al., 2016) propose a communication architecture for Body Area Networks (BANs) and design a scheme to secure the data communication between wearable sensors and data consumers (doctors and nurses). They propose the CP-ABE and signature-based schemes to store the encrypted data at the data sink. Alrawais et. al (Alrawais et al., 2017) propose a key exchange protocol based on ciphertext policy attribute-based encryption (CP-ABE) to secure communications in fog-enabled IoT systems. The security properties namely confidentiality, authentication, verifiability and access control, in such an environment are achieved by combining CP-ABE with digital signature techniques.

Moreover, some studies also propose proxy re-encryption schemes whereby fog nodes re-encrypt the ciphertexts and/or keys for end devices. Wang et. al (Wang, 2018b) propose an ID-based proxy re-encryption scheme to secure the communication between end devices and cloud. The proposed scheme is leakage resilient in auxiliary input model. It follows a hybrid encryption approach wherein the data files are encrypted using symmetric keys, whilst the symmetric keys are encrypted with the master public keys. Whenever an end device wants to access a file it sends a request to cloud which subsequently sends the encrypted symmetric

key to fog nodes. Upon receiving the encapsulated ciphertext, the fog nodes re-encrypt the key to generate the decryption key for the end device.

Jiang et. al (Jiang et al., 2018) investigate the property of key-delegation abuse in ABE systems. A provably secure CP-ABE scheme against key-delegation abuse is proposed. A new security game model against key-delegation abuse is also introduced. The new feature of the proposed CP-ABE scheme is proved in a generic group model. An application of traitor tracing CP-ABE scheme is also presented.

Diro et. al (Diro et al., 2018) propose a proxy re-encryption scheme based on elliptic curves for securing communication between fog nodes and IoT devices. In the proposed scheme, the computationally heavy cryptographic operations are offloaded to fog nodes which act as broker. The fog nodes convert a message $m$ encrypted under the public key $PK_1$ of the receiver to another ciphertext using another public key $PK_2$ without revealing the contents of the message and private keys. The proxy re-encryption scheme guarantees data confidentiality.

To the best of this researcher's knowledge, there is no trust computation model and/or TMS for Fog-CPS. As the Fog-CPS shares many commonalities with cloud computing (Alhanahnah et al., 2018; Fan and Perros, 2014; Ghosh et al., 2015; Li et al., 2015; Nagarajan and Varadharajan, 2011; Xiaoyong et al., 2015), Internet of Things (IoT) (Namal et al., 2015) (Nitti et al., 2014), wireless sensor networks (WSN) (Li and Song, 2017), and mobile adhoc networks (MANETs) (Shabut et al., 2015; Wang et al., 2014; Xia et al., 2016), the models proposed for these systems are somewhat relevant.

The majority of studies compute trust in cloud services on the basis of objective trust but some adopt a hybrid approach (Alhanahnah et al., 2018; Fan and Perros, 2014; Ghosh et al., 2015; Lu and Yuan, 2018; Nagarajan and Varadharajan, 2011; Xiaoyong et al., 2015) where trust is the fusion of objective and subjective evidence. The literature identifies two popular methods to compute objective trust, a) subjective logic (Josang and Ismail, 2007), and b) real-time adaptive trust evaluation approach (Lu and Yuan, 2018) (Li et al., 2015) (Xiaoyong et al., 2015). In adaptive trust evaluation approaches, the trust computation problem is modeled as a process of multi-attribute decision making (MADM) and weights are assigned adaptively either by information entropy (Lu and Yuan, 2018) (Li et al., 2015) or maximizing deviation method (Xiaoyong et al., 2015); whereas in subjective logic weights are assigned manually or subjectively (Josang and Ismail, 2007).

Recently, Lu et. al (Lu and Yuan, 2018) adopted a TOPSIS approach (Technique for Order of Preference by Similarity to Ideal Solution) for evaluating the trustworthiness of cloud service by combining objective and subjective evidence. TOPSIS is a multi-criteria decision analysis method. Somu et. al (Somu et al., 2018) propose a hypergraph binary fruit

fly optimization based service ranking algorithm (HBFFOA) for selection of trustworthy cloud services.

Tian et. al (Tian et al., 2017) propose a trust evaluation approach for sensor-cloud systems. In this work, trust evaluation issue is formulated as a multiple linear regression (MLR) problem. Wang et. al (Wang et al., 2014) proposed a logistic regression based trust Model for MANETs. Soleymani et. al (Soleymani et al., 2017) employ fuzzy logic based on experience and plausibility to establish trust among vehicles in vehicular ad hoc networks (VANETS). In this work, fog nodes are entrusted with the task to evaluate the correctness of the information received from authorized vehicles. Likewise, the study in (Wang et al., 2018b) also employs fog computing to evaluate trustworthiness in sensor-cloud systems. It proposes a hierarchical trust computation mechanism to compute trust in the sensor network and between sensor service providers and cloud service providers. Moreover, for trust computation in CPS systems, many different approaches namely trusted computing (Rein et al., 2016), game theory (Pawlick and Zhu, 2017), and generic probabilistic graph modelling (Wang, 2018a) have been proposed.

## 1.2 Research Motivation

Having discussed the related studies for the Fog-CPS systems the discussion now moves to their limitations and the need to design a new secure integrated framework.

The frameworks discussed in the previous section have the following limitations. Firstly, the cited studies have addressed these issues separately. They either address the security or trust challenges, no study follows an integrated approach. Secondly, the security frameworks based on graph modelling techniques do not achieve the fundamental security goals, namely data confidentiality, integrity, authentication and authorization. The same is the case with the key agreement frameworks, they also cannot enforce authentication and authorization. Thirdly, no trust computation framework is proposed for Fog-CPS systems. Fourthly, in existing trust models, the evaluation is partial. Trust is either computed for IoT devices, sensor nodes or cloud services.

Furthermore, the cited encryption schemes also have a number of limitations. Symmetric key schemes are suitable for resource constrained devices due to their smaller key sizes. However, when short-size data is encrypted with a symmetric key, then the information which is revealed about the key may be critical for ciphertext-only attack. Moreover, in large-scale Fog-CPS systems, for instance, smart grid, the symmetric key management process becomes very complicated and complex. Symmetric schemes also require a separate protocol for session key agreement, distribution, and generation.

Furthermore, existing attribute based encryption (ABE) and public key encryption (PKE) (Bethencourt et al., 2007; Chen et al., 2011; Goyal et al., 2006; Zhou et al., 2015) also have a number of limitations. Firstly, in PKE schemes, the generation, verification, and distribution of certificates incur extra computation and communication overhead. Secondly, in existing ABE schemes, there is a CA which generates secret keys. However, the compromise of CA can endanger the secret keys and therefore the secrecy of encrypted messages. Thirdly, in existing ABE schemes, all the attributes belonging to a user attribute set are used in key generation. The privacy of attributes is compromised in case of leakage of secret keys. Fourthly, ABE schemes which are based on bilinear pairing are computationally complex and require large security parameters (i.e. 1024 or 2048-bit size). Fifthly, in ABE schemes the access structures and/or policies have one-to-many association between attributes and keys. In other words, they are based on a one-to-many relationship where many secret keys are generated for a single set of attributes. All parties whose secret keys satisfy a specific access policy can decrypt the ciphertext. Sixthly, the existing schemes focus on the key establishment, authentication, and authorization.

Next, the limitations of existing trust computation models are discussed. Despite having well established trust models for cloud computing, they cannot be directly applied to fog scenarios due to the decentralized and distributed architecture of fog-enabled systems.

Most of the existing studies on IoT, MANETs, and CPS only evaluate the trustworthiness of sensor nodes and CPS devices. However, as the fog nodes can be deployed in open and unprotected environments, they are also vulnerable to cyber attacks and can therefore be compromised. Nonetheless, as discussed above it is essential to evaluate the trustworthiness of all Fog-CPS entities.

It is underlined that similar to other distributed systems namely P2P, MANETs, and sensor clouds, the Fog-CPS systems are also vulnerable to self-promotion, bad-mouthing, opportunistic service, on-off, collusion, Sybil, and ballot-stuffing attacks. These attacks aim to degrade the accuracy of the trust computation model or impact the availability of TMS itself. For instance, in a self-promotion attack, attackers attempt to increase their own trust by reporting false parameters. A bad-mouthing attack occurs when a node gives bad recommendations about other nodes. In the case of a Fog-CPS system, malicious CPS devices can send false parameter reports regarding their experience with fog nodes to purposefully decrease their trust.

Additionally, fog nodes can be opportunistic at times meaning that they will provide good services only for their own benefit. Similar to opportunistic service, in on-off attacks, malicious entities can behave in ways that are good and bad depending upon the situation. Likewise, in collusion attacks, several compromised CPS devices can collaborate to modify

the trust results of other Fog-CPS entities. In a Sybil attack, a malicious node (i.e. CPS devices in Fog-CPS systems) can create several fake IDs to report false values of trust parameters. Moreover, in Fog-CPS systems, a ballot stuffing attack occurs when a CPS device submits more parameter reports than permitted in a given time period with the aim of attackers being able to affect the trust of fog nodes.

These attacks can result in imprecise trust computation which does not reflect the true actions and/or performance of Fog-CPS entities. It is therefore essential to devise counter-measures against these attacks such that the trust cannot be maliciously manipulated.

## 1.3 Overview of the Approach

To address the limitations of existing encryption schemes and trust computation models, this research work proposes a novel integrated security framework which is an enabler of trustworthy and dependable fog-based systems. The proposed framework is comprised of two fundamental components, 1) a security component (SC) and 2) a trust management system (TMS). The SC guarantees that all entities of Fog-CPS, i.e. fog nodes and cyber physical system (CPS) devices, have unique identities and only authorized parties can access the fog resources. The TMS evaluates the trustworthiness of Fog-CPS systems by computing trust for each of the entities. Trust computed by TMS is further used in access control policies to guarantee that only trusted entities can request fog services and collaborate with other entities in the system. The integration of SC and TMS ensures that security and trust challenges are adequately addressed.

As part of SC, a novel lightweight encryption scheme based on elliptic curves is proposed to enforce robust authentication and authorization. ECC schemes are more efficient since they use smaller key sizes (i.e. 128 or 256-bit) and are therefore more suitable for resource limited Fog-CPS systems. The motivation comes from the IBE and ABE schemes. This researcher believes that inherent properties/attributes of CPS devices can be used for identification, authentication, and authorization. All CPS devices from a single sensor to a fog node are characterized by a set of attributes e.g. identities, interactions, types of data (stored and processed), and locations etc. These inherent attributes of CPS devices are considered as an initiative to create cryptographic keys. The scheme presented here combines the properties of IBE, ABE, and asymmetric encryption schemes as it uses attributes to generate the public/secret key pair.

Moreover, in the proposed scheme, the CA cannot decrypt the messages exchanged among CPS devices thus maintaining their privacy. This is achieved by dividing the attribute set into two parts namely secret and shared attributes. The secret attributes are only shared

with CA which registers the entities and publishes the public keys. The registration of CPS entities with CA ensures that published partial public keys are authentic and do not need any further verification. However, as the secret attributes are only known to CA, the other collaborating CPS devices cannot verify them. To address this problem, a notion of shared/public attributes is introduced. The shared attributes are known to collaborating Fog-CPS entities which generate the final public and secret keys.

The encryption and decryption would take place using the final public and secret keys. Such an approach is advantageous for two reasons, 1) the secret attributes are only shared with the CA and the leakage of secret keys would not risk the communication of collaborating entities and 2) the scheme is scalable because the final public keys are generated by the collaborating devices themselves without the aid of CA. Any device can generate the final public keys of other devices. Additionally, this scheme eliminates the need to have separate constructions for authentication and authorization, as only those devices which possess a shared attribute set can collaborate. Furthermore, upon key revocation, the key regeneration process is lightweight, since we do not repeat the whole key generation process.

Considering the limitations of existing trust computation models, this work proposes a holistic TMS for Fog-CPS systems. The proposed TMS handles trust computation, management and dissemination. TMS evaluates the trustworthiness of Fog-CPS entities by monitoring their behaviour based on quality of service (QoS) and network communication features. Trust is computed for each fog node and CPS device, and is readily available for other entities to access. Based on the trust, other devices can make a decision to collaborate.

Moreover, the proposed trust computation model includes several components for trust computation and credibility evaluation of fog nodes and CPS devices. A major advantage of the credibility evaluation is the prevention of Sybil, collusion, and data anomalies attacks. TMS ensures the functionality of the Fog-CPS system is not compromised such that it produces unintended outcomes.

Additionally, the integration of SC and TMS is achieved by embedding trust as an attribute in the access control policies in the proposed Fog-CPS security scheme in SC component. The access policies defined based on trust ensure that the entities with a desirable trust score can decrypt the message and therefore collaborate with other entities in the system. Integrating the system behaviour indicators (i.e. trust parameters) with security properties would enable a multi-factor and multi-dimensional trustworthiness evaluation of Fog-CPS systems. This researcher believes such an approach can alleviate the risk and uncertainty associated with interactions in open and distributed Fog-CPS environments.

## 1.4 Research Question, Objectives and Hypotheses

### 1.4.1 Research Question

How can trustworthiness in fog-enabled cyber physical systems (Fog-CPS) be improved?

### 1.4.2 Research Objectives

The objectives of this study are enumerated below:

1. Research Objective 1: To investigate the security and trust challenges of Fog-CPS systems.
2. Research Objective 2: To integrate security and trust into a framework for the Fog-CPS systems.
3. Research Objective 3: To propose an efficient and lightweight security component which addresses security challenges namely data confidentiality, authentication, and authorization.
4. Research Objective 4: To design a resilient and accurate trust management system (TMS) for Fog-CPS systems.

### 1.4.3 Research Hypotheses

To achieve the research objectives following research hypotheses have been defined:

1. Research Hypothesis 1: It would be possible to investigate the security and trust challenges of Fog-CPS systems by reviewing related literature including but not limited to cloud-assisted CPS systems, fog computing architectures, IoT systems and cloud computing.
2. Research Hypothesis 2: It would be possible to integrate the security and trust in a security framework. The integration would enable multi-faceted and multi-dimensional trustworthiness evaluation of Fog-CPS systems.
3. Research Hypothesis 3: An efficient and lightweight encryption scheme based on elliptic curves can achieve data confidentiality, authentication and authorization by employing attributes belonging to Fog-CPS systems for establishing identities and access control.

4. Research Hypothesis 4: A resilient and accurate trust management system (TMS) can be designed to evaluate the trustworthiness of all Fog-CPS entities. It would also be possible to protect the TMS against malicious attackers by incorporating some trust credibility metrics.

## 1.5 Contribution

The main contributions of this thesis are as follows:

- **Investigation of Security and Trust Challenges in Fog-CPS.** This is the first part of the research in which several CPS and fog computing systems were studied in order to investigate their security, trust and privacy issues.
- **Developing an Integrated Security Framework.** An integrated security framework was developed by following a cross-property trustworthiness and risk assessment approach proposed by NIST (NIST, 2017). However, this research only considered security and trust.
- **Proposing an Efficient and Lightweight Security Component (SC).** A lightweight encryption scheme based on elliptic curves was designed to address the security challenges of Fog-CPS systems.
- **Designing of a Resilient and Accurate TMS.** A resilient and accurate TMS was developed to compute the trust of all entities in a Fog-CPS system. The TMS complements the SC system to address the security challenges and serves as a countermeasure for malicious behaviour of Fog-CPS entities.

## 1.6 Thesis Outline

- **Chapter 1 Introduction:** This chapter provides information on the context of the research in hand, together with the scope, aims and objectives, and the scientific contribution of the research.
- **Chapter 2 Literature Review:** Chapter 2 throws more light on fog computing including its architecture, features, and the integration with CPS. Moreover, it presents related integrated frameworks. It also discusses the state of the art in security schemes and security evaluation techniques. It further reviews the existing literature on trust models.

- **Chapter 3 Secure Integrated Framework:** Chapter 3 presents the proposed secure integrated framework. It also presents a generalized architecture of a fog-enabled smart power grid control system (Fog-SGC). It further discusses the security and trust challenges specific to a Fog-SGC system.
- **Chapter 4 Experimental Evaluation and Results:** Chapter 4 presents the results of the proposed SC and TMS systems.
- **Chapter 5 Conclusion and Future Work:** Chapter 5 concludes this thesis with a summary of its key achievements, challenges and open ended research questions which may be relevant to future research studies.

## 1.7 Publications

Parts of this dissertation have been published (Komninos and Junejo, 2015) and submitted to reputed journals.

**Published**

- N. Komninos and A. K. Junejo, "Privacy Preserving Attribute Based Encryption for Multiple Cloud Collaborative Environment," 2015 IEEE/ACM 8th International Conference on Utility and Cloud Computing (UCC), Limassol, 2015, pp. 595-600.

**Submitted**

- A. K. Junejo, N. Komninos, M. Sathiyanarayanan and B. S. Chowdhry, "Trustee: A Trust Management System for Fog-enabled Cyber-Physical Systems", IEEE Transactions on Emerging Topics, November 2018.

# Chapter 2

# Literature Review

This chapter presents some background information related to CPS systems, fog computing architecture and its features. Next, the related studies are presented. Following that, the related integrated frameworks are reviewed. The state of the art in the security schemes is also presented. Moreover, the trust computation models related to CPS, IoT, MANETs and fog computing are discussed. Lastly, the enhancement of security techniques with trust is discussed.

## 2.1 Fog Computing in Cyber Physical Systems

In Chapter 1, fog computing and Fog-CPS systems are briefly introduced. In this section, fog computing architectures are discussed in detail. As a further matter, the features of fog computing which make it well suited for CPS systems are briefly introduced. OpenFog Consortium Architecture Working Group (2017) defines fog computing as follows:
**Definition 1:**

*"A horizontal, system-level architecture that distributes computing, storage, control and networking functions closer to the users along a cloud-to-thing continuum."* (OpenFog Consortium Architecture Working Group, 2017)

Fog computing architectures can broadly be classified into two categories (Ni et al., 2018):

1. Cloud-Fog-Device Framework
2. Fog-Device Framework

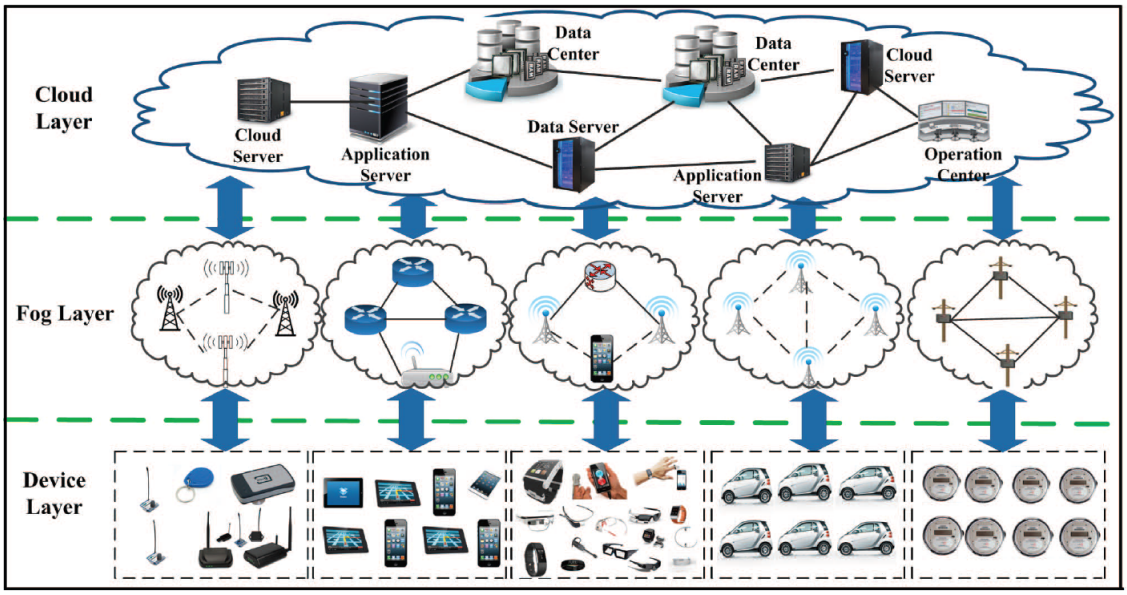


Fig. 2.1 Three-Layer Architecture of Fog Computing (Ni et al., 2018)

### 2.1.1 Cloud-Fog-Device Framework

This framework consists of three distinct layers, namely, the *device*, *fog* and *cloud*, as illustrated in Fig. 2.1. The three layers of "Cloud-Fog-Device" framework are organized in an increasing order of computing and storage capabilities (Challa et al., 2017). Each entity is equipped with wired (e.g., Ethernet, optical fiber), wireless (e.g., Bluetooth, LTE, ZigBee, NFC, IEEE 802.11 a/b/c/g/n, satellite links) or a combination of both communication technologies to achieve inter layer and cross layer communications (Sehgal et al., 2015). Fog computing is a highly virtualized platform in which different virtualization technologies, such as network functions virtualization and software-defined networks (Vaquero and Rodero-Merino, 2014a) are used to achieve network virtualization and traffic engineering.

Similar to cloud computing, each of the above mentioned layers is scalable, meaning that more CPS devices, fog nodes and cloud resources can be added to cope with the increasing demand and load balancing. Moreover, similar to other distributed systems, these layers can be connected with trusted third parties i.e. certification authorities (CA), key generation authorities (KGA) and arbitrators to generate keys, issue certificates and handle threats in case of some accidents and frauds. These trusted anchors introduce some initial level of trust in fog computing based systems and enable a dependable and trustworthy environment.

### Device Layer

The *device layer* has two types of CPS devices, namely mobile and fixed. The mobile devices such as wearables (i.e. smart watches, smart clothes, smart glasses, fitness trackers) and mobile (i.e. smart phones and autonomous vehicles) are carried by the owners (Sehgal et al., 2015). All devices belonging to the same owner, deployed at same place, and/or belonging to the same company or organization can form a group and communicate with each other using wireless ad hoc networks. Fixed CPS devices (e.g. sensors and RFID tags) are pre-deployed in specific places to fulfill pre-defined tasks (e.g. traffic monitoring, water level checking, products tracing, forest fire detection, smoke detection and air quality monitoring). The CPS devices monitor the physical environments by collecting the sensory and contextual data and subsequently reporting it to fog nodes. Large scale fog-enabled systems such as smart cities would have thousands of CPS devices installed at several places to monitor and collect data from different subsystems of future cities.

The CPS devices are resource constrained and cannot execute computationally heavy operations (Gazis, 2017). Moreover, they cannot immediately respond to emergency situations.

### Fog Layer

The *fog* layer consists of computational, storage and networking equipment commonly known as fog nodes in the fog computing model (Ni et al., 2018). The fog nodes extend the cloud services closer to the network edge and may be up to the CPS devices, IoT sensors and actuators. The *fog* layer provides transient storage and real-time analysis on the data collected by CPS devices. Depending upon a specific use case, in *fog* layer, there could be routers, bridges, gateways, switches and base stations, augmented with computational capability and local servers (e.g., industrial controllers, embedded servers, mobile phones and video surveillance cameras) (Ni et al., 2018). The fog nodes are not completely fixed to the physical edge, but should be seen as a fluid system of connectivity. Additionally, regarding the fog networking, the study in (Chiang, 2015) underlines a few important aspects. It is argued that fog networking constitutes two planes, namely, control plane and data plane. The control plane is responsible for configuration and management of the network (e.g., sync routing table information and efficient traffic control) Whereas the transfer of data from source to destination is the responsibility of the data plane. The routing information is provided by the control plane logic.

#### Cloud Layer

The *cloud* layer in the Cloud-Fog-Device framework provides compute, storage and networking services to cyber-physical applications from a global perspective. The cloud delivers on-demand computing resources to users and/or CPS devices over the internet such that they can access it from anywhere at any time. The multi-tenancy and virtualization technologies allow the cloud resources to be shared among multiple users. Several CPS applications can independently and concurrently run on the cloud.

Furthermore, the *cloud* layer hosts SaaS applications for the acquisition, processing, presentation, and management of the sensory data. In the three-layer framework, the fog nodes provide intermediary computational and storage services closer to the network edge where the data is being generated. Such an approach enables real-time data processing and quick decision making. However, the computationally intensive and data dense operations related to advanced analytics and business intelligence (BI) in CPS applications (Inc., 2015), such as smart power distribution (Jalali et al., 2016), health status monitoring (Bonomi et al., 2012) and network resource optimization (Peng et al., 2016) are executed in the *cloud* layer. In addition, the cloud also sends policies to the fog layer to improve the quality of latency-sensitive services offered by fog nodes.

#### Fog-Device Framework

In the Fog-Device framework, there are two layers, namely the *device* and *fog*. In this framework, several fog nodes collaborate with each other to offer services to CPS devices. The Fog-Device framework can be employed in use cases such as decentralized vehicular navigation (Ni et al., 2016), indoor floor plan reconstruction (Chen et al., 2015a), smart traffic lights (Shropshire, 2014) and local content distribution (Ahmed and Rehmani, 2017).

### 2.1.2 Features of Fog Computing

The key features of fog computing are discussed below.

- **Location awareness:** The location of fog nodes can be traced actively or passively to support devices with rich services at the network edge (Shropshire, 2014). As the fog nodes are located in the vicinity of CPS devices, the regions in which CPS devices are operating is known. Subsequently, the information regarding the local network condition, mobility pattern of users and precise location information can easily be retrieved.

- **Availability:** Fog computing is more reliable than its counterpart traditional cloud computing which suffers from a variety of connectivity problems. The higher reliability guarantees higher availability of fog systems.
- **Ubiquity:** The ubiquitous presence of CPS devices and fog nodes ensures dependable real-time applications in fog-enabled systems.
- **Context Awareness:** Fog nodes have greater context-awarenesses as they are located in the proximity of CPS devices.
- **Geographic Distribution:** Unlike cloud servers which are located at remote places, the fog nodes are deployed locally to cover wider geographic areas. The fog nodes can be deployed at specific positions, such as along highways and roadways, on cellular base stations, on a museum floor and at a point of interest (Shropshire, 2014). The near-to end devices fog nodes can process the data streams in real-time thus ensuring latency-sensitive applications in Fog-CPS.
- **Low latency:** Large-scale connected systems are generating data at an exponentially growing rate. Sending all data to the cloud will result in performance and network congestion challenges. Besides that, mission-critical use cases require real-time data analysis because only then subsequent actions can be beneficial. The fog computing paradigm fits well in scenarios where data will be processed within fog nodes and response should be generated with low latency (Shropshire, 2014).
- **Local Resource Pooling:** In fog-enabled systems, the computational resources (e.g., processing, memory and communication) are provisioned from locally available fog nodes.
- **Large-Scale CPS Application Support :** As mentioned in Section 1.1, 500 billion devices will be connected to the Internet and will generate massive amounts of data. Cloud computing cannot handle this exponential growth of data and service requests. Owing to geographical distribution, local resource pooling, context and location awareness features, fog computing can easily support large-scale cyber-physical applications such as smart cities, environment monitoring, power grid management, water treatment management and climate change monitoring. Moreover, fog computing is scalable and can easily manage billions of CPS devices (Bonomi et al., 2012).
- **Decentralization:** Fog computing is a decentralized architecture such that there is no centralized server to manage resources and services. The fog nodes self-organize

to cooperatively provide real-time services and applications to users (Vaquero and Rodero-Merino, 2014b).

- **End Device Mobility:** End devices can be fixed and mobile as well. Fixed devices are pre-deployed as part of the infrastructure whereas the mobile devices (e.g. smart phone, smart watch, smart glasses and smart vehicles) are owned by individuals and their location varies with user mobility.
- **Heterogeneity:** A very distinct and challenging feature of fog computing is the heterogeneity. The CPS devices and fog nodes are heterogeneous by design with varying degrees of computation and storage capacity.
- **Edge Analytics and Stream Mining:** The near-user fog nodes are the key enablers of enhanced user experience with reduced latency for quality of service (QoS).
- **Capacity of Processing High Number of Nodes:** The fog computing architecture is highly scalable as fog nodes can be added and removed any time in the network. Owing to these qualities, fog nodes can serve a large number of end devices and applications.
- **Wireless Access:** Most of the entities in fog computing are equipped with wireless connectivity.
- **Real-time Applications:** Fog computing is an appropriate and viable platform for real-time applications supporting fast data analysis and follow up actions.

Fog-enabled solutions are more suitable for CPS as the services with reduced latency, network jitter and response time are located near the edge of network (OpenFog Consortium Architecture Working Group, 2017). The above features and/or advantages of fog computing are particularly important for mission-critical, data-dense, and latency-sensitive use cases, such as power grid, health-monitoring, surveillance, gaming and video streaming where a massive amount of data is being generated in order to solve the limitations in current infrastructures.

### 2.1.3 Fog Computing Standards

Having presented the architecture and key features of fog computing, in this section the endeavours of Open Fog Consortium to devise a reference architecture for fog computing are discussed.

The OpenFog Consortium is a joint venture of big technology companies and academic institutions namely, ARM, Cisco, Dell, Intel, Microsoft Corporation and the Princeton

University Edge Laboratory. It was established on November 19, 2015. The OpenFog Consortium aims to standardize and promote fog computing in various fields across numerous disciplines. Recently, on 13 February 2017, the OpenFog Consortium released its Reference Architecture (RA) (OpenFog Consortium Architecture Working Group, 2017). The RA marks a remarkable first step towards creating an open architecture to enable high-performance, scalable, interoperable and secure large-scale, data-intensive and complex CPS systems. The OpenFog RA outlines deployments models for different types of systems ranging from embedded to fully interconnected large clustered systems. Two major types of deployment models include:

1. Hierarchical Deployment
2. Multi-tier Fog Deployment

Moreover, the RA also proposes fog deployment models for smart traffic systems, food processing chains, smart cities and airport visual security use cases.

## 2.2 Integrated Frameworks

Having elaborated on several aspects of fog computing, the related studies are now presented. Integrated frameworks are discussed first and subsequently the security schemes and trust models are reviewed.

Mayer et. al (Mayer et al., 2017) present an open semantic framework (OSF) that enables the users to create intelligent services by using all available resources. The study proposes the use of semantic technologies namely ontologies of interlinked terms, concepts, relationships, and entities to add meaning to the communication among different IoT devices. Sicari et. al (Sicari et al., 2017) advocate the need to define dynamic policies to manage access control and information distribution in IoT. A distributed middleware is proposed to disseminate policies to different heterogeneous devices in an IoT network. The policies are applied to the communication protocols and information shared between IoT devices. The correct functioning of the middleware is guaranteed by integrating it with a synchronization system which in turn enables the real-time distribution, update and enforcement of the policies.

Alippi et. al (Alippi and Roveri, 2017) underline that the current solutions proposed for the CPS systems are inadequate to fulfil the dynamic service requirements coming from either users or applications. To address these challenges, the study proposes an information-centric approach that enables the development of homogeneous and harmonized smart CPS systems. The proposed information centric adaptive systems (INCAS) framework is a layered architecture comprised of approximate computing, fault diagnosis, energy management and

learning in nonstationary environments. Mohsin et. al (Mohsin et al., 2017) propose a probabilistic model checking framework to formally and quantitatively analyze the risks in different configurations of IoT use cases. The proposed model takes an IoT configuration, vulnerability scores and the capabilities of the attacker as input. It subsequently generates the threat models to compute the attack likelihood and attacker cost for the specific IoT configuration under consideration.

Hu et. al (Hu et al., 2017) propose a security and privacy preserving scheme to address the challenges arising in a fog-based face identification and resolution framework. The face identification approaches are widely being employed to establish identity management in IoT. The proposed scheme employs fog computing to decrease the latency and processing resources. In order to guarantee a reliable and dependable cloud service in a cloud-enabled IoT environment, Chen et. al (Chen and Zhu, 2017) propose a contract based FlipCloud game. The game assesses the quality of service (QoS) and security risks involved in the interactions between IoT devices and the cloud under persistent threats. The game incorporates a pricing mechanism for on-demand security as a service for cloud-based IoT systems. Similar to (Chen and Zhu, 2017), the work in (Han et al., 2018) evaluates the security level provided by a given cloud service under consideration in a cloud-based IoT system. The study proposes a security assessment framework based on software defined networks. The three-layer framework is designed by integrating the software defined networks and cloud IoT systems. It consists of 23 different indicators to describe the security features. The authors claim that the proposed framework can effectively evaluate the security level of a cloud IoT system under consideration. The work in (Yin et al., 2018) detects and mitigates DDoS attacks by employing the software-defined anything (SDx) network technology. The proposed framework enables the secure management of IoT devices. It consists of a controller pool containing SD-IoT controllers, SD-IoT switches integrated with an IoT gateway, and IoT devices.

Khalid et. al (Khalid et al., 2018) presents a security framework for industrial CPS systems in the context of industry 4.0. The application is demonstrated in a human-robot collaborative system. The study comprehensively outlines the basic elements, and functional requirements of a secure collaborative robotic CPS. It further describes the attack models for such a system. The proposed security framework is based on a two-pronged strategy wherein the data security is guaranteed at important interconnected adapter nodes. The work in (DiMase et al., 2015) proposes a systems engineering framework for cyber physical security and resilience. It is maintained that security can be guaranteed by ensuring resilience of CPS systems. Such an approach will enable the application of both integrated and targeted security measures and policies in critical infrastructure.

The work in (Hahn et al., 2015) presents a multi-layered security analysis framework for elaborating the cyber attacks and risks faced by CPS systems. The study further explores the various notions related to attacks particular to CPS systems, namely attacker objectives, cyber exploitation, control-theoretic properties and physical system properties. Additionally, the proposed framework outlines the progressive stages of attacks to highlight the steps required for an attacker to launch a successful attack against a CPS system. The study in (Sarigiannidis et al., 2017) proposed an analytic framework for modelling security attacks in IoT systems. The proposed framework employs dynamic G-network theory to model the attacks. Furthermore, the positive arrivals denote the data streams that originated from the various data collection networks (e.g., sensor networks), while the negative arrivals denote the security attacks that result in data losses (e.g., jamming attacks). The intensity of an attack is categorized as high and low depending upon the damage and/or losses caused.

The study in (Rathore et al., 2018) proposed a framework to select a robust security service in a fog and mobile-edge computing environment. A soft hesitant fuzzy rough set (SHFRS) approach is employed to solve multi-criteria decision making problems. SHFRS approach is an integration of hesitant fuzzy rough set theory and hesitant fuzzy soft set.

Due to the decentralized nature of blockchain, it is being widely incorporated to address various problems across diverse disciplines. Likewise, it is deemed as the most essential cyber security technology to address the security and privacy challenges faced by emerging computing paradigms, namely cloud-enabled IoT systems, CPS systems and industrial IoT systems. The study in (Puthal et al., 2018) explains the key characterstics of the blockchain technology and further underlines its applications, namely financial services, IoT and smart health care etc. Rahman et. al (Rahman et al., 2018) propose a blockchain based mobile edge computing secure therapy framework. The framework uses blockchain Tor-based distributed transactions to preserve the therapeutic data privacy, ownership, generation, storage, and sharing. Moreover, the edge computing paradigm is employed to reduce the bandwidth, latency and processing time.

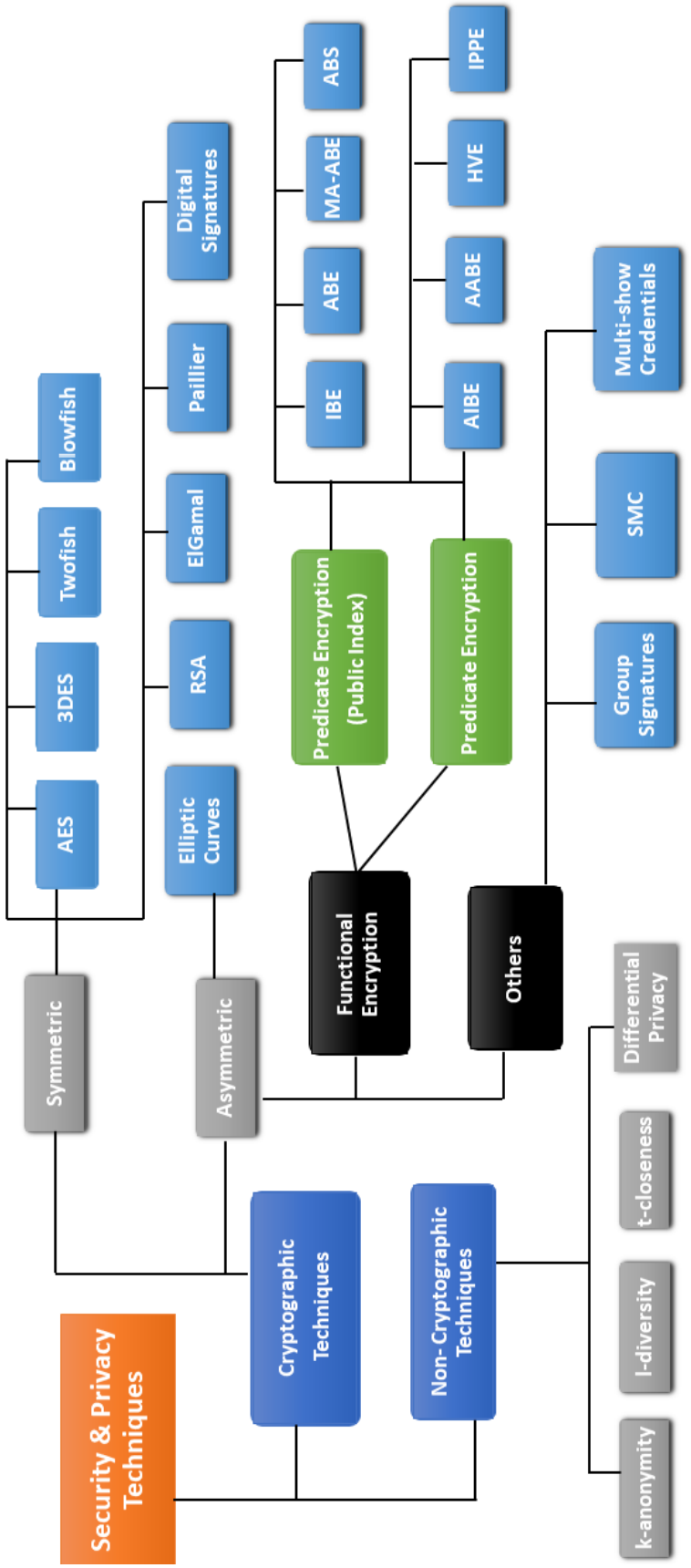


Fig. 2.2 Classification Tree for Security and Privacy Technologies

## 2.3 Security Schemes

The literature is full of cryptographic and non-cryptographic techniques on authentication, authorization and anonymization, proposed for various domains and application scenarios. Figure 2.2 illustrates the classification tree for existing security and privacy techniques. The security and privacy enhancing technologies can broadly be classified as:

- Cryptographic Techniques
- Non-Cryptographic Techniques

### 2.3.1 Cryptographic Techniques

The use of encryption techniques for securing open systems and ensuring confidentiality, integrity, authentication and authorization has long been in practice. Encryption schemes can be divided into three main types:

- Symmetric Key Encryption (Private Key)
- Asymmetric Key Encryption (Public Key)
- Cryptographic Hash Functions (No Key Algorithms)

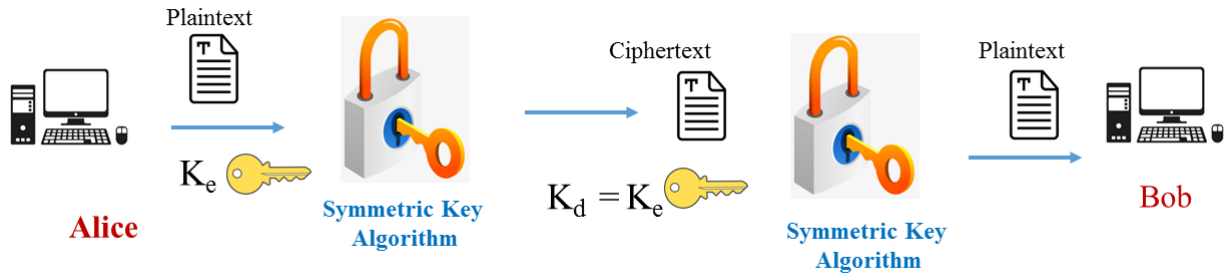


Fig. 2.3 Symmetric Key Cryptography Based Communication System

### 2.3.2 Symmetric Key Encryption

In this type of encryption, same key is used for both the encryption and decryption . Figure 2.3 shows the symmetric key based communication between Alice and Bob. It can be observed that same key is being used for encryption and decryption i.e. $K_e = K_d$. If Alice wants to send a message to Bob encrypted using symmetric key encryption, then the key used

for encryption must also be sent to Bob through some secure channel. Otherwise the key exchange should take place before encryption. AES, 3DES, Twofish, and Blowfish are some of the popular symmetric key encryption algorithms. In these schemes, the key is distributed via a secure channel. With regard to security, the symmetric key encryption provides higher level of security than asymmetric key encryption per bit. However, symmetric key encryption has some disadvantages as listed below:

- Firstly, key distribution is a major problem as all parties need to have the same key.
- Secondly, the key management becomes quite difficult in large scale systems. Furthermore, as the keys are changed frequently it also contributes to key management problems.
- Thirdly, in some cases, the symmetric encryption cannot achieve authentication and non-repudiation.

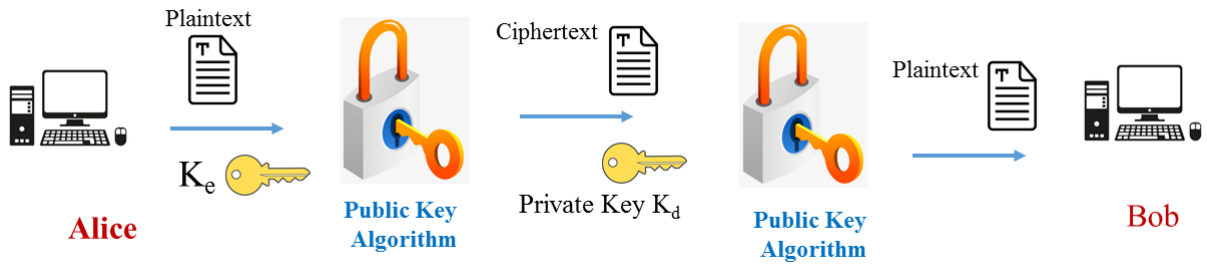


Fig. 2.4 Public Key Cryptography Based Communication System

### 2.3.3 Asymmetric Key Encryption

In early 1970s, Whitfield Diffie and Martin Hellman (Diffie and Hellman, 1976) proposed the notion of public key cryptography. Asymmetric or public key encryption (PKE) solved some of the fundamental problems of cryptography, namely key distribution, key exchange and key management. In comparison to symmetric encryption, asymmetric encryption also provides confidentiality, non-repudiation and authentication. Additionally, asymmetric encryption involves several mechanisms, namely key exchange protocol, digital signature algorithms and encryption. Soon after, the PKE schemes become the key techniques to secure data transmission in distributed networks and/or systems.

In PKE, instead of one, two separate keys, namely public and private, are used for encryption and decryption. As the name suggests, the public key is known to all while the

private key is only known to a specific user. Anyone can use the public key to encrypt a message but only the user who possesses the private key can decrypt it. Figure 2.4 illustrates the public key based communication between Alice and Bob. This time two different keys i.e. $K_e$ and $K_d$ are used for encryption and decryption. Alice uses Bob's public key $K_e$ to encrypt a message for him. However, Bob uses its private key $K_d$ to decrypt the ciphertext.

As shown in Fig. 2.2, there are different PKE schemes, namely RSA, ElGamal, Paillier, functional encryption, digital signatures and group signatures etc. All asymmetric encryption schemes have their roots in Diffie-Hellman (D-H) key exchange protocol (Diffie and Hellman, 1976) or the RSA cryptosystem (Rivest et al., 1978). The D-H key agreement protocol employs cyclic groups with special properties. However, the RSA encryption does not require the cyclic groups but uses similar arithmetic operations. Additionally, in RSA, the order of the group is not known to the attacker.

#### Security of Public Key Cryptography

The security of PKE systems is based on following problems:

- Discrete Log Problem
- Computational Diffie-Hellman Problem (CDHP)
- Decisional Diffie-Hellman (DDH) Problem

But before discussing the above mentioned mathematical problems in details, the notion of cyclic groups is introduced.

#### Cyclic Groups

In abstract algebra, a cyclic group $G$ is a group that can be generated by a single element $g$, called group generator. Other elements of the group $G$ can be generated from the generator $g$ for some integer values $i$ based on the form $g^i$.

#### Implementation of Cyclic Groups

In abstract algebra, cyclic groups can be implemented based on two mathematical constructions, namely finite fields and elliptic curves. Both of these are briefly discussed below but elliptic curves will be explored in more detail later.

### Finite Fields

In order to generate a finite field, first, a large prime number $p$ is chosen and then a subgroup $G_1$ of $\mathbb{Z}_p^*$ of prime order $q$ is chosen. However, for RSA crytosystems, $p$ is chosen such that it is the product of two large prime numbers i.e. $R$ and $S$. In such cases, when the order of $p$ is not known, the computations still takes place in $\mathbb{Z}_p^*$.

### Elliptic Curves

For elliptic curves, one takes an elliptic curve $E$ over some finite field $F_q$ and takes some subgroup $\mathbb{G}$ of the group of points $E(F_q)$ with prime order $p$, so the group operation is point addition.

For elaborating above problems, following mathematical notations are used. Let $\mathbb{G} = \langle g \rangle$ is a cyclic group of prime order $p$. Let $g$ be a generator of $\mathbb{G}$. Let $a$, $b$, $c$ be integers in $[0, p-1]$.

### Discrete Log Problem (DLP)

DLP is a classical mathematical problem in cryptography. It is defined as follows:
Given a cyclic group $\mathbb{G}$, a generator $g$ and another element $h$ of $\mathbb{G}$, the problem is to find the discrete logarithm to the base $g$ of $h$ in the group $\mathbb{G}$. Or in other words, given the group generator $g$, an element $g^a$, the problem is to compute $a$.

### Computational Diffie-Hellman Problem (CDHP)

In CDHP, the problem is to find a new group element given two group elements. For example, given the group generator $g$ of group $\mathbb{G}$, two group elements $g^a$, $g^b$, the problem is to compute $g^{ab}$.

### Decisional Diffie-Hellman (DDH) Problem

In DDH problem, the task is to find the secret numbers without performing the actual computations. Precisely, given $g$, and three other group elements i.e. $g^a$, $g^b$, $g^c$, the problem is to determine if $ab = c$. Majority of the cryptographic schemes are designed based on the difficulty of solving one of these hard mathematical problems. If DLP can be solved for a given group then all other problems such as CDHP and DDH can also be solved.

### Diffie-Hellman Key Agreement

Diffie-Hellman (D-H) key agreement is the first PKE protocol to share a secret key among multiple parties. With this method, the shared secret key can be securely distributed over a public network. There are two types of Diffie-Hellman key agreement protocols, namely, two-party D-H and three-party D-H.

### Two-Party D-H Key Agreement

In two-party key agreement, the secret key is shared between two parties, $A$ and $B$. For secret key sharing, both the parties need to agree on a cyclic group $\mathbb{G}$ of prime order $p$ having a group generator $g$. Once both the parties agree on group parameters, the secret key is shared based on following steps:

1. $A$ chooses a secret number $a$ from a finite field $\mathbb{Z}_p$, likewise $B$ also chooses a secret number $b \in \mathbb{Z}_p$.

2. $A$ then computes $g^a$ and sends it to $B$, in a similar fashion, $B$ computes $g^b$ and sends to $X$.

3. Next, $A$ computes $K_B = (g^b)^a = g^{ab}$, and $B$ computes $K_A = (g^a)^b = g^{ab}$.

4. Finally, both $A$ and $B$ have established the secret key $K = K_A = K_B$ without revealing their respective secrets, $a$ and $b$.

### Three-Party D-H Key Agreement

As the name suggests, in three party D-H key agreement protocol, there are three parties, $A$, $B$ and $C$ who intend to share a secret key. The secret sharing among three parties cannot be achieved by following the two-party method, as $A$ cannot easily incorporate the public key of $C$ to generate the shared key $g^{abc}$. To share the secret among three parties, bilinear mapping is used. In bilinear mapping, the group elements of two cyclic groups are used to yield an element of a third group:

$$e : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T \tag{2.1}$$

where $\mathbb{G}_T$ is a target group with $\mid \mathbb{G}_T \mid = \mid \mathbb{G} \mid = p$. Using bilinear mapping, the secret among three people is shared based on following steps:

1. Firstly, $A$, $B$ and $C$ each select a secret number $a$, $b$, $c \in \mathbb{Z}_p$ respectively.

2. Next, $A$, $B$ and $C$ compute $g^a$, $g^b$ and $g^c$ respectively and subsequently broadcast them.

3. Following that $A$ computes $e(g^b, g^c) = e(g,g)^{bc}$ and then raises this to his secret $a$ to compute the shared secret $K = (e(g,g)^{bc})^a = e(g,g)^{abc}$.

4. Lastly, $B$ and $C$ will also follow the same procedure in order to compute $K$.

#### Identity Based Encryption

Identity based encryption (IBE) is a type of the public key cryptography wherein any string (i.e. email address) can be a valid public key. Unlike classical public key cryptography, with IBE schemes, the public keys do not need to be distributed beforehand. Boneh and Franklin's IBE scheme is perhaps the most famous early example of what could be achieved using bilinear maps (Lynn, 2007a), though not the first (Sakai et al., 2000), (Joux, 2000). Apart from IBE, bilinear mapping is used in several other applications, namely attribute based encryption, attribute based signatures and anonymous credentials.

### 2.3.4 Cryptographic Hash Functions

Apart from symmetric and asymmetric encryption schemes, hash functions are also considered as the fundamental pillars of the modern cryptography. The hash functions takes as input a string of any length but output is always of a fixed size. SHA1 (Gallagher, 2012), SHA2, SHA3 (Morris, 2015) (recommended by NIST), MD4 and MD5 (Rivest, 1992) are some of the hash functions.

For example, Alice wants to send a message to Bob. She calculates hash value of the message using a hash function. She then encrypts the hash value using asymmetric cryptography algorithm which is called a form of digital signature. Alice also creates an arbitrary session key for symmetric encryption. The key is used for encryption of the message. The private key is encrypted using public key of Bob using public key cryptography. Now a digital envelope is formed, including the message and encrypted session key. Alice then sends the digital envelope and digital signature to Bob. Bob retrieves the session key using his private key. Finally, the message of Alice is decrypted with the help of symmetric key algorithm using the session key. Alice also decrypted the hash value using the Alice public key to verify integrity. Bob uses the decrypted message to generate a hash value using hash algorithm and compare with the value of decrypted hash value. The hybrid procedure ensures Bob several goals such as private message (symmetric encryption), the message is only for Bob ( Bob's private key used to decrypt), the message is not be altered (by matching hash value) and Alice sent the message(Alice public key is used to generate the same hash value).

## 2.4 Elliptic Curve Cryptography

Elliptic curve cryptography is also a type of public key cryptography. Both Koblitz (Koblitz, 1987) and Miller (Miller, 1986) proposed elliptic curve cryptography in 1985. ECC is based on algebraic structures of elliptic curves which are defined over finite fields. Due to achieving higher security levels with shorter key sizes, ECC has widely been used in many applications. Moreover, an added advantage of ECC is that the user has the freedom to choose a number of parameters. ECC is based on the Discrete Logarithm Problem (DLP). An elliptic curve $E$ over finite field $F_q$ is defined by equation 2.2:

$$E : y^2 = x^3 + ax + b \tag{2.2}$$

where $q = p^m$ is a prime power, $(p \neq 2,3)$ is a prime integer and $m$ is some positive integer. Moreover, the coefficients $a$ and $b$ are secret numbers chosen from finite field $F_q$ such that $4a^3 + 27b^2 \neq 0$. Each elliptic curve point is represented as an ordered pair $(x,y)$ such that it satisfies Equation 2.2. The coordinates $x$ and $y$ of each elliptic curve point are elements of the finite field $F_q$. Each elliptic curve $E$ has a special point $\infty$ which is an identity of the group. A characteristics property of point $\infty$ is that if any point $P$ on $E$ is added to it then:

$$P + \infty = P \tag{2.3}$$

Besides that, two point $P$ and $Q$ on an elliptic curve can always be added to get another point. In other words, addition of points is an elliptic curve operation in the group $\mathbb{G}$. When the point addition operation is performed underlying arithmetic operation in the finite field is called field arithmetic operation (Khan, 2015). However, to perform, scalar multiplication in the elliptic curves, a point $P$ is added $k$ times to get a new point $Q$. Equation 2.4 computes the scalar multiplication of point $P$.

$$Q = kP = P + P + .... + P. \tag{2.4}$$

where $k = log_P Q$ is called discrete logarithm problem of $Q$ to the base point P. Moreover, the inverse operation, i.e. calculating $k$ in Equation 2.4, is again a mathematical problem called Elliptic Curve Discrete Logarithm Problem (ECDLP). ECDLP is a harder problem than DLP. Up until today, no algorithm can solve ECDLP in sub-exponential time.

Regarding the security of ECC schemes, the elliptic curves provides a higher security per bit than its counterparts asymmetric schemes. Table (Lenstra and Verheul, 2001) lists the key sizes for symmetric, asymetric and elliptic curves.

| Symmetric | RSA/ DSA/ Diffie-Hellman | Elliptic Curve |
|---|---|---|
| 80 | 1024 | 163 |
| 112 | 2048 | 233 |
| 128 | 3072 | 283 |
| 192 | 7680 | 409 |
| 256 | 15360 | 571 |

Table 2.1 Key Sizes for Symmetric and Asymmetric Cryptosystems (bits) (Lenstra and Verheul, 2001)

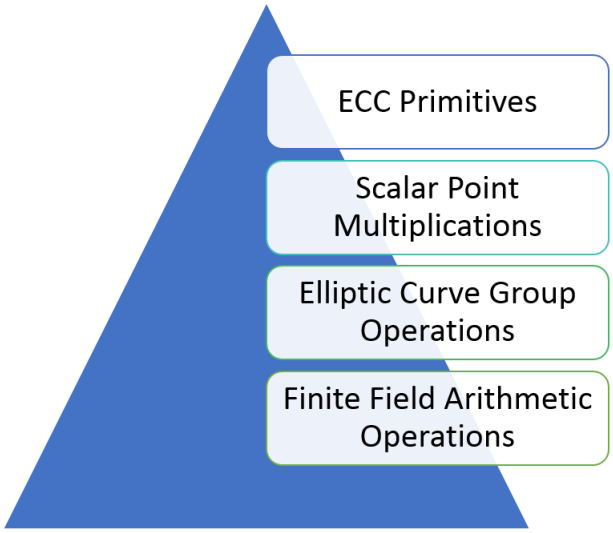


Fig. 2.5 Hierarchy of ECC Operations (Khan, 2015)

### 2.4.1 Scalar Point Multiplication

Scalar point multiplication is the key operation of the ECC. It is a computationally expensive operation and therefore encryption and decryption takes most of the time in any ECC scheme. Figure 2.5 illustrates the hierarchy of elliptic curve cryptography operations. The elliptic curve operations, namely point addition and point doubling are employed to perform scalar point multiplication. However, the above mentioned point operations are based on the finite arithmetic operations as shown in Figure 2.5.

### 2.4.2 Finite Field Theory

Modern cryptography is based on finite field theory, developed by French mathematicians Evariste Galois. For last few decades, the finite theory has widely been used in cryptographic schemes. (Menezes et al., 1996) defines finite fields as follows:

"In the finite field theory, a finite field, $F_q$ or a Galois field, $GF$ is an algebraic structural group of finite numbers. The algebraic operations such as addition , subtraction, multiplication with and division are performed within elements of the finite field while it is maintaining algebraic laws such as associative, commutative, distributive, existence of an additive identity is 0 and a multiplicative identity is 1, additive inverse, and multiplicative inverse for nonzero elements. The group structure of $GF$ also follows a Group law. For example, a group $M$ is called a commutative group or Abelian group. Modern cryptography systems are based on the Abelian groups (Menezes et al., 1996)."

### Finite Fields in Cryptography

Generally in cryptography following three fields are used (Khan, 2015):

- If $m = 1$, then the $GF(p)$ is called prime field.
- If $m \geq 2$, then the $GF$ is called extension field.
  - If $p = 2, m > 1$, then the $GF(2^m)$ is called a binary extension field or characteristic 2 field or simply binary field.
  - If $p > 2$, $m > 1$, then the $GF(p^m)$ is called an optimal extension field.

### Representations of Finite Fields

Elliptic curves can be implemented in three ways i.e. by using the basis of $GF(p^m)$ namely:

1. Polynomial (Canonical/standard) basis
2. Normal basis
3. Dual basis

The carry free arithmetic operations in polynomial basis makes them appropriate for hardware implementation of ECC. However, in software implementation of the elliptic curves, the normal basis and dual basis are used as the squaring operation can be performed by a simple shift operation.

### Domain Parameters of Elliptic Curve Cryptography

For each cryptosystem, a group of parameters are designed in order to protect it against all malicious attacks. Likewise, for elliptic curves, a set of parameters are designed such that users can use them to securely communicate with each other. The parameters discussed herein are taken from (Khan, 2015). For more details, the interested readers are encouraged to

read this dissertation. The ECC domain parameters $(q, FR, a, b, P, p, h)$ are further classified into prime field domain parameters and binary field domain parameters (Schneier, 1993) (Montgomery, 1987) where:

- $q$ = Order of the Field,
- $FR$ = the field representation of any element under $GF(q)$,
- $a$, $b$ = Two field elements define the base point of an elliptic curve over a finite field for a field characteristic,
- $P$ = A base point $(x, y) \in E/GF(q)$ where $x$ and $y$ are the coordinates,
- $n$ = A prime number as an order of the $P$ or key length of ECC is a significant parameter for security,
- $h =$ Cofactor, $\#E_q/n$ where $\#E_q = nh$; $n$ is a prime number and $h = 1, 2, 3, 4$ is an integer.

The term, $\#E_q$ is chosen a prime or almost prime and $n \geq 160$ to avoid Pohlig-Silver-Hellman and Polar-$\rho$'s methods based solution of the discrete logarithm problem (Ian et al., 1999), (Julio and Ricardo, 2000) and (Koblitz et al., 2000).

### 2.4.3 Elliptic Curve Cryptography Protocols

ECC has four main protocols for key generation, D-H key exchange and digital signatures as enumerated below:

1. Elliptic Curve Key Generation
2. Elliptic Curve Diffie-Hellman Key Exchange (ECDH)
3. ElGamal Elliptic Curve Cryptosystem
4. Elliptic Curve Digital Signature Algorithm (ECDSA)

## 2.5 Pairing Based Cryptography

Another important application of elliptic curve cryptography is pairing based cryptography. Most of the material discussed in this section is taken from (Moody et al., 2015). The NIST report (Moody et al., 2015) comprehensively discusses the various aspects related to pairing based cryptography.

Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be two additive cyclic groups of prime order $p$, then a pairing between elements of these groups outputs another point in a target group $\mathbb{G}_T$. Pairing can formally be represented using Equation 2.5:

$$e : \mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_T \tag{2.5}$$

Additionally, the bilinear pairing is computed using Equation 2.6:

$$e(P_1 + P_2, Q) = e(P_1, Q) * e(P_2, Q), \tag{2.6}$$

where the $*$ denotes multiplication in the finite field. The bilinear mapping preserves the additive structure of the elliptic curve, and carries it over into the finite field. The bilinearity property opened up many new cryptographic applications. Majority of the recent functional encryption schemes are based on bilinear pairing. Elliptic curves are generally written using Weierstrass equation:

$$y^2 = x^3 + ax + b. \tag{2.7}$$

Nowadays, research is being conducted to compute pairing on other models of elliptic curves including:

1. Huff curves:
$$x(ay^2 - 1) = y(bx^2 - 1) \tag{2.8}$$
2. Jacobi quartics:
$$y^2 = ex^4 - 2dx^2 + 1 \tag{2.9}$$
3. Twisted Edwards curves:
$$(a)x^2 + y^2 = 1 + dx^2y^2 \tag{2.10}$$

### 2.5.1 Bilinear Maps

Let $E$ be an elliptic curve defined over a finite field $F_q$. Let $P$ and $Q$ be points of order $r$ on $E$, where $Q$ is generally defined over an extension field of $F_q$. Let $\mu_r$ be the group of $r^{th}$ roots of

unity in $F_q^k$ computed using Equation 2.11:

$$\mu_r = \left\{ \alpha \in F_q^k : \alpha^r = 1 \right\}. \tag{2.11}$$

Pairing is subsequently defined as:

$$e : \langle P \rangle \times \langle Q \rangle \to \mu_r. \tag{2.12}$$

Precisely, pairing maps two points on an elliptic curve $E$ to a finite field $F_q$.

### 2.5.2 Properties of Pairing

A pairing must satisfy following three properties:

1. **Bilinearity:** Every pairing must be bilinear:

$$\begin{aligned} e(P_1 + P_2, Q) &= e(P_1, Q)e(P_2, Q), \\ e(P, Q_1 + Q_2) &= e(P, Q_1)e(P, Q_2) \end{aligned}$$

2. **Non-degenerate:** The pairing must be non-degenerate, meaning that the pairing is not trivial. This property is only satisfied if $e(P,Q) \neq 1$, for two elliptic curve points $P$ and $Q$.

3. **Computable:** The pairing must be efficiently computable.

The embedding degree $k$ is the smallest integer such that $r \mid (q^k - 1)$. Alternatively, $k$ is the order of $q$ mod $r$. The value of the pairing is an element of the finite field $F_q^k$. In order for the pairing to be efficiently computable, $k$ must be **small**, certainly less than 100.

### 2.5.3 Decisional Bilinear Diffie-Hellman problem (DBHP)

Three-party Diffie-Hellman key agreement lends itself to the decisional form, DBDH. Like DDH, DBDH requires a participant to determine if some target element is either a special combination of given parameters or a random element:

- Given $g$, $g^a$, $g^b$, $g^c \in \mathbb{G}$ and $T \in \mathbb{G}_{\mathbb{T}}$.
- Determine if $T = e(g,g)^{abc}$ or a random element, where $e$ is a bilinear map $\mathbb{G} \times \mathbb{G} \to \mathbb{G}_{\mathbb{T}}$.

This problem is more difficult than three-party D-H key agreement since the "secret keys" $a$, $b$, and $c$ are unknown.

### 2.5.4 Security of Pairing-based Cryptography

Much of elliptic curve cryptography relies on the difficulty of two problems known as the *Discrete Log Problem (DLP)* and the *Computational Diffie-Hellman Problem (CDHP)*. These problems have been well studied, and if parameters are correctly chosen then they are believed to provide adequate security (Moody et al., 2015). The security assumption behind **pairing-based cryptography** is known as the **Bilinear Diffie-Hellman Problem (BDHP)**. It is known that if one can solve the *DLP* or *CDHP* then one can also solve the *BDHP*. So, the security of pairing-based cryptography is not stronger than that of elliptic curve cryptography. There are no currently known attacks to break the *BDHP*, and it is the focus of much research (Moody et al., 2015).

### 2.5.5 Pairing-Friendly Elliptic Curves

Although in theory pairings exist for any elliptic curve, in practice there are curves whose pairings are not suitable for cryptographic applications. Associated to each elliptic curve, there is a parameter that can be calculated known as the embedding degree $k$. In order to efficiently implement pairings for use in cryptography, we need $k$ to be relatively small, certainly less than 100. However, it has been shown that almost all elliptic curves have very large $k$. In fact, $k$ is usually about the same size as $q$, which is at least 160 bits (Moody et al., 2015).

### 2.5.6 Types of Pairing-Friendly Curves

There are two types of pairing-friendly curves:

1. Supersingular Curves
2. Ordinary Curves

**Supersingular Curves**: The first example of pairing-friendly curves is supersingular curves. A supersingular elliptic curve always has embedding degree $k \leq 6$.

**Ordinary Curves:** Curves which are not supersingular are called ordinary. With overwhelming probability, a randomly chosen elliptic curve will be ordinary. There are various families of pairing-friendly ordinary curves, all of which are constructed by using what is known as the complex multiplication (CM) method (Moody et al., 2015).

### 2.5.7 Common Ways to find Pairing-Friendly Curves

There are two common ways to find pairing-friendly elliptic curves (Moody et al., 2015):

- The first is to use what are known as supersingular elliptic curves, which always have $k \leq 6$. A supersingular elliptic curve over $F_q$ is a curve with $q+1$ points. It has been proven that supersingular curves always have embedding degree $k \leq 6$. The Voltage Security incorporation which is cofounded by Dan Boneh recommends using the following supersingular curve

$$y^2 = x^3 + b \text{ over } F_q, \tag{2.13}$$

  where $q$ is a prime with $q \equiv$ 11 mod 12.

- The second way is to use a technique called the complex multiplication (CM) method to construct certain families of elliptic curves with small $k$.

#### Complex Multiplication (CM) Method

The curves which are not supersingular are called ordinary. In many cases, a randomly selected elliptic curve would be ordinary. However, complex multiplication method can be used to construct pairing-friendly ordinary curves. Before describing the CM, Hasse's theorem which gives the tight bounds on the cardinality of $E(F_q)$ is discussed. Let $N$ be the number of points on $E$, $q$ is the order of the finite field $F_q$ and $t = q+1-N$. By Hasse's theorem, it is known that $t \leq 2\sqrt{q}$. Hence, $N = q+1-t$ where $\mid t \mid \leq 2\sqrt{q}$. If $q \mid t$ then $E$ is a supersingular curve otherwise $E$ is ordinary. If $|t| \leq 2\sqrt{q}$ and $q \nmid t$, then there exists an elliptic curve $E$ over $F_q$ with $E(F_q) = q+1-t$. Let $D = 4q-t^2$ (Moody et al., 2015). Informally, the CM method consists of the following steps:

1. Construct Hilbert class polynomial $H_D(x)$ (possible for $D \leqslant 10^{13}$);
2. Find a root j (mod p) of $H_D(x)$;
3. Create an elliptic curve $E$ with $j - invariant\, j$;
4. Check $E$ and its twist $\acute{E}$ for a point of large prime order $r$.

The trick to using the CM method is how to find values for the parameters q, t, D, and r so that k is small. There are many families of ordinary curves given in (Freeman et al., 2010).

### 2.5.8 Commonly used Pairings

The two most commonly used pairings are the Weil (Miller, 2004) and Tate (Matsuda et al., 2007) pairings. The Weil pairing satisfies $e(P,P) = 1$ for any point $P$ in the domain, while the other pairings do not. With the goal of speeding up computation, researchers have discovered several new pairings including:

- Ate (Hess et al., 2006)
- Eta (Lee et al., 2009)
- Reduced Tate (Matsuda et al., 2007)
- Twisted Ate (Matsuda et al., 2007)
- R-Ate (Lee et al., 2009)

### 2.5.9 Choice of Pairing

There is no single pairing choice that is the all-around best. It depends on the specific protocol, security level needed and curve choice, etc. Several such considerations are now explored. For most protocols, if a formal proof of security is required, then supersingular curves must be used. Supersingular curves also have the advantage of using distortion maps to change the domain from to $\langle P \rangle$ to $\langle Q \rangle$

$$\langle P \rangle \times \langle Q \rangle \rightarrow \langle P \rangle \times \langle P \rangle .$$

The **Eta pairing** can only be defined over supersingular curves.

#### Types of Pairing

Pairings are categorized into three types (Freeman et al., 2009). Each pairing can be written as $e : \mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_T$, where the domain is $\mathbb{G}_1 \times \mathbb{G}_2$. Typically the points in $\mathbb{G}_1$ have coordinates in $F_q$, while those in $\mathbb{G}_2$ have coordinates in $F_q^k$.

1. **Type 1:** The pairing $e$ is Type 1 if $\mathbb{G}_1 = \mathbb{G}_2$
2. **Type 2:** The pairing $e$ is Type 2 if $\mathbb{G}_1 \neq \mathbb{G}_2$, and there is an efficiently computable homomorphism $\varphi : \mathbb{G}_2 \rightarrow \mathbb{G}_1$ (but not vice versa).
3. The pairing $e$ is Type 3 if $\mathbb{G}_1 \neq \mathbb{G}_2$, and there are no efficient homomorphisms between $\mathbb{G}_1$ and $\mathbb{G}_2$.

Table 2.2 from (Galbraith et al., 2008) illustrates some of the differences between pairings. A checkmark ✓ denotes the pairing can easily achieve the property, while an $x$ denotes it cannot. Let $p$ be the characteristics of $F_q$, i.e., $q = p^f$

Table 2.2 Comparison of Pairings

| Type | Hash to $\mathbb{G}_2$ | Short $\mathbb{G}_1$ | Homomorphism | Poly time Generation |
|---|---|---|---|---|
| 1 (p = 2 or 3) | ✓ | × | ✓ | × |
| 1 (p >3) | ✓ | × | ✓ | ✓ |
| 2 | × | ✓ | ✓ | ✓ |
| 3 | ✓ | ✓ | × | ✓ |

- Hash to $\mathbb{G}_2$: One can hash into $\mathbb{G}_2$
- Short $\mathbb{G}_1$: There is a (relatively) short representation for elements of $G_1$
- Homomorphism: There is an efficiently computable $\varphi : G_2 \leftarrow \mathbb{G}_1$
- Poly time generation: One can generate system parameters (including groups and a pairing) achieving at least $K$ bits of security in time polynomial in $K$.

## 2.6 Pairing-based Encryption Schemes

This section discusses a few security and privacy schemes which are based on pairing based cryptography.

- Functional Encryption (FE) Schemes
- Anonymous Credentials System

### 2.6.1 Functional Encryption Schemes

Functional encryption (FE) schemes provide fine-grained access control over encrypted data. These schemes are widely used to protect the user data in clouds. There are different FE schemes that provide multiple levels of security and privacy. The functional encryption is a generalized form for most of the recent and popular encryption schemes such as identity based encryption (IBE), attribute based encryption (ABE) and predicate encryption (PE) schemes. The aforementioned schemes are derived from two sub-classes of FE. The two classes differ on the types of predicates which are applied on the plain-text messages (Boneh et al., 2011).

- Predicate Encryption with Public Index
- Predicate Encryption (PE)

### 2.6.2 Predicate Encryption with Public Index

The IBE and various forms of ABE schemes fall into the first class of predicate encryption schemes. These schemes allow for expressive forms of access control, they are limited in two ways. First, the access policy is given in the clear which is often in itself can be considered sensitive. Second, it does not allow for computation on the encrypted data, which includes applications such as search. The PE schemes with public index are discussed below:

1. Attribute based Encryption (ABE)
   (a) Distributed Multi-Authority ABE
2. Attribute based Signature (ABS)

### 2.6.3 Attribute based Encryption Schemes

In an attribute based encryption (ABE) scheme, the initiator of communication, or sender, defines an access policy over a set of attributes such that the users who satisfy the policy can only decrypt the ciphertext. There are two subcategories of ABE schemes:

1. Cipher-text Policy ABE (CP-ABE)
2. Key-Policy ABE (KP-ABE)

#### Ciphertext policy Attribute based Encryption

In Ciphertext policy ABE (CP-ABE) (Bethencourt et al., 2007), ciphertext defines access policy and user keys define attributes. The decryption takes place only if the attributes of user key match with the attributes of the access policy.

#### Key-Policy policy Attribute based Encryption

In Key-Policy ABE (KP-ABE) (Goyal et al., 2006), ciphertexts are associated with sets of descriptive attributes, and user keys are associated with policies. The ciphertext policy is more flexible because the users who can decrypt the message do not need to be known beforehand. In Key-Policy ABE, the access policy is associated with the keys and only those users who are in possession of keys can decrypt.

### 2.6.4 Distributed Multi-Authority ABE Schemes

Lewko et. al (Lewko and Waters, 2011) propose a Multi-Authority Attribute-Based Encryption (ABE) system. In this scheme, any party can become an authority and there is no requirement for any global coordination other than the creation of an initial set of common reference parameters. A party can simply act as an ABE authority by creating a public key and issuing private keys to different users that reflect their attributes. A user can encrypt data in terms of any boolean formula over attributes issued from any chosen set of authorities. Finally, the proposed system does not require any central authority.

The proposed decentralized attribute based encryption scheme does not require any central authority. It avoids the performance bottleneck incurred by relying on a central authority, which makes the system more scalable. It also avoids placing absolute trust in a single designated entity which must remain active and uncorrupted throughout the lifetime of the system. This is a crucial improvement for efficiency as well as security, since even a central authority that remains uncorrupted may occasionally fail for benign reasons, and a system that constantly relies on its participation will be forced to remain stagnant until it can be restored. In the proposed scheme, authorities can function entirely independently, and the failure or corruption of some authorities will not affect the operation of functioning, uncorrupted authorities.

### 2.6.5 Attribute Based Signature Scheme

Maji et. al (Maji et al., 2011) introduce Attribute-Based Signatures (ABS), a versatile primitive that allows a party to sign a message with fine-grained control over identifying information. In ABS, a signer, who possesses a set of attributes from the authority, can sign a message with a predicate that is satisfied by his attributes. The signature reveals no more than the fact that a single user with some set of attributes satisfying the predicate has attested to the message. In particular, the signature hides the attributes used to satisfy the predicate and any identifying information about the signer (that could link multiple signatures as being from the same signer). Furthermore, users cannot collude to pool their attributes together.

An ABS scheme which uses RSA-like operations and keys is proposed in (Herranz, 2014). Herranz (Herranz, 2014) proposes a lightweight RSA-style ABS scheme. This scheme employs RSA operations for signing. The scheme supports threshold signing policies $(t,n)$ where the signer is verified if he possesses some $t$ out of $n$ attributes. The scheme is proved secure under Random Oracle Model.

### 2.6.6 Predicate Encryption (PE)

The second class of Predicate Encryption systems do not leak the access policy and achieves both payload (message) hiding and attribute hiding.

- Anonymous Identity-Based Encryption (Anonymous IBE)
- Anonymous Attribute Based Encryption
- Hidden Vector Encryption (HVE)
- Inner Product Predicate(IPP)

### 2.6.7 Anonymous Attribute Based Encryption

The Anonymous (Hidden) CP-ABE schemes preserve the privacy of receivers by hiding the policy attributes with wild-cards or asterisks ($*$). The studies in (Phuong et al., 2016; Sreenivasa Rao and Dutta, 2015; Zhou et al., 2015) propose anonymous schemes. Similar to other FE schemes, the receiver has access to plain-text if he satisfies the policy.

### 2.6.8 Predicate Encryption Scheme

The ABE schemes discussed above hide the message but the access policy is publicly known. The predicate encryption (PE) achieves better privacy preservation than FE as it hides both the message (payload) and the attribute set. Predicate encryption schemes additionally provide anonymity as ciphertexts also conceal the attribute set they are associated with, which is known to enable efficient searches over encrypted data. Lewko et. al (Lewko et al., 2010) propose a fully secure construction of predicate encryption. It supports different types of predicates such as equality, conjunctive, and disjunctive.

### 2.6.9 Hidden Vector Encryption Scheme

Hidden Vector Encryption (HVE) is a type of predicate encryption where the attribute set associated with the ciphertext or the user secret key can contain wildcards (De Caro et al., 2013). HVE supports the fine-grained conjunctive combination of equality, comparison, and subset queries on encrypted data.

### 2.6.10 Functional Encryption Schemes Based on Elliptic Curves

In the preceding sections, different types of FE schemes are discussed. The PE schemes with public index, namely ABE and ABS, are employed to enforce access control in cloud services. They are also useful in environments where there are several data creators and data collectors/analyzers. The data creators, i.e. the users, can encrypt the data based on an access policy defined over a set of attributes. The receivers can decrypt the data if and only if they have corresponding secret keys.

However, a major limitation of existing predicate schemes, more specifically ABE, is the underlying computational complexity. The bilinear mapping or pairing operations require large security parameters i.e. 1024 or 2028 bits. These schemes are not suitable for resource constrained devices with limited storage and processing powers.

Recently, some ABE schemes (Challa et al., 2017; Hu et al., 2016; Mahmood et al., 2018; Odelu and Das, 2016; Yaoa et al., 2015a) based on elliptic curves have been proposed. Due to the unique characteristics of elliptic curves, such schemes are lightweight having much smaller security parameters and key sizes compared to their counterpart public key encryption schemes.

### 2.6.11 Discussion on Cryptographic Schemes

After reviewing the literature, it can be concluded that all of the above mentioned FE schemes can achieve fine-grained access control and authorization. The IBE schemes are less practical for Fog-CPS, since they only deal with a single attribute. ABE schemes are more appropriate for defining flexible access policies. Moreover, the multi-authority schemes are built upon the idea of distributed and/or decentralized authorities. Such schemes are scalable, efficient, and dependable, therefore well-suited for large-scale cyber physical systems.

The ABS scheme is a promising approach to authenticate a signer based on a set of attributes. The verifier does not gain any further information than the fact that some user who possesses the required attributes has signed the message. The ABS schemes preserve the privacy of the senders.

Hidden vector encryption (HVE) schemes (De Caro et al., 2013) are more suitable for applications where there is a need to search or do computation over the encrypted data as in the case of cloud servers which provide storage services and grant access to the data if certain access policies are satisfied. The encrypted data can be searched with the help of a trap door (token) generated to the cloud server by the data owner. The token is comprised of several predicates. The HVE schemes increase the computational cost and requires several operations to decrypt the messages even for the simplest of tasks.

Furthermore, the inner product encryption scheme achieves both attribute hiding and payload hiding. Having discussed different cryptographic schemes their limitations are now outlined:

1. In public key encryption schemes, the generation, verification, and distribution of certificates incur extra computation and communication overhead. Resource constrained CPS devices do not have sufficient computing, storage, and energy powers for asymmetric schemes.

2. Due to small key sizes, symmetric key schemes are well suited for resource constrained devices. However, in large-scale fog-enabled systems, for instance, smart traffic system, smart grid, and smart cities, the symmetric key management process becomes very complicated and complex. Symmetric schemes require a separate protocol for session key agreement and generation. Moreover, when short-size data is encrypted with a symmetric key, then the information which is revealed about the key may be critical for ciphetext-only attack.

3. In existing ABE schemes, there is a CA which generates the secret keys. However, the compromise of CA can endanger the secret keys and therefore the secrecy of encrypted messages.

4. The ABE and asymmetric schemes described above require large security parameters (i.e. 1024 or 2048-bit size) for bilinear pairing. Such schemes are not suitable for systems with resource constrained CPS devices. ECC schemes are more efficient, since they use smaller security sizes (i.e. 128 or 256-bit).

## 2.7 Security Evaluation Techniques

This section briefly discusses the well established security notions, attacks and properties of cryptographic schemes.

### 2.7.1 Provable Security

A cryptographic scheme is considered to be provably secure if the security of the scheme can be proved by a mathematical proof. In a provable security model, the capabilities of the attacker are defined by an adversarial model. In order to break the security of the scheme, the attacker must solve an underlying hard mathematical problem. The provable security model does not consider side channel attacks or other implementation related attacks. According to (Hevia, 2013), the provable security consists of the following steps:

- Define goal of the proposed scheme and/or adversary
- Define attack model
- Design a protocol
- Define complexity assumptions (or assumptions on the primitive)
- Provide a proof by reduction
- Verify proof
- Interpret proof

### 2.7.2 Security Models

There are four security models namely:

#### Random Oracle Model

In cryptography, a random oracle is referred to as a theoretical black box which responds to every unique query with a (truly) random response chosen uniformly from its output domain. The random oracle (RO) model is defined on the assumption of the existence of an ideal hash function which should return a uniformly distributed random output for each unique input. Many encryption schemes are designed on the mathematical abstractions of the RO model. The main problem with the RO model is that it is nearly impossible to build such a model. Moreover, a secure hash function should be resilient to collisions, preimages and second preimages attacks. These properties do not imply that the function is a random oracle.

#### Standard Security Model

In the standard model, the security of an encryption scheme is based on the hardness of mathematical problems, namely discrete log problem (DLP) and bilinear Diffie-Hellman (BDH). This model does not make any assumptions about random functions.

#### Selective Security Model

The selective security model is based on a weaker security notion. However, it is different from the standard model, as this model dictates the interaction between the adversary and the challenger. In the selective model, the adversary has to declare in advance the types of challenges it will get in the security game and/or protocol. For example, in ABE schemes, the adversary submits the access policy to the challenger before getting the secret key.

#### Adaptive Security Model

In contrast to the selective security model, in the adaptive model, the adversary can adaptively change the submitted challenges and then follow the protocol accordingly.

### 2.7.3 Cryptanalysis Attack Models

Cryptanalysis is the study of information systems with the aim of anaylyzing and investigating their hidden aspects and subsequently finding and improving techniques for breaking or weakening them. In cryptanalysis, the ciphertext, ciphers, and cryptosystems are examined methodically to retrieve and/or gain access to the contents of encrypted messages, even if the secret key is unknown. In cryptanalysis, the attacks are categorized based on the type of information accessible to the attacker. Following are some main attack models:

- Ciphertext-only Attack(COA): In this type of attack, the cryptanalyst has access only to a number of ciphertexts and there is no way the attacker can get access to plaintext before encryption.
- Known-plaintext Attack (KPA): As the name suggests, in this attack model, the attacker has access to both a set of plaintext messages and their corresponding ciphertexts.
- Chosen-plaintext Attack (CPA): The attacker can obtain the ciphertexts corresponding to any plaintexts of his choice.
- Adaptive Chosen-plaintext Attack: This attack model is similar to a chosen-plaintext attack, however, the attacker can choose subsequent plaintexts based on information learned from previously obtained ciphertexts.
- Chosen-ciphertext Attack (CCA): Like chosen-plaintext attack, chosen-ciphertext attacks, can also be adaptive or non-adaptive. In non-adaptive CCA, the cryptanalyst chooses ciphertexts in advance and obtains the decryptions, i.e. plaintexts, under an unknown key. However, the attacker should not inform the choice of more ciphertexts based on already obtained plaintexts. There is a special variant of CCA namely lunchtime or midnight attack.
- Adaptive Chosen-ciphertext Attack (CCA2): In adaptive chosen-ciphertext attack, the adversary also has access to a decryption oracle which (adaptively) decrypts any ciphertext of his choice except one specific ciphertext (called the challenge).

- Related-key attack: Like a chosen-plaintext attack, except the attacker can obtain ciphertexts encrypted under two different keys. The keys are unknown, but the relationship between them is known; for example, two keys that differ in the one bit.

### 2.7.4 Security Notions for Cryptographic Schemes

The two important security properties of cryptographic schemes including both encryption and signature based schemes are:

- Indistinguishability
- Non-malleability

#### Indistinguishability

In encryption schemes, the security goal is to achieve "Perfect Secrecy". However, it is not possible to achieve "Perfect Secrecy", because information theoretically ciphertext also reveals some information about the plaintext. Considering the difficulties of achieving the notion of "Perfect Secrecy", the security goal is relaxed to "Indistinguishability" or "Semantic Security". Ciphertext indistinguishability is an important security property of many encryption schemes. Indistinguishability is defined as follows:

*"Given the ciphertext and the encryption key, the adversary cannot tell apart two same-length but different messages encrypted under the scheme, even if chose the messages himself."*

Depending upon the capabilities of the attacker, indistinguishability can be defined in several ways. It is normally presented as a security game, where the cryptosystem is considered secure if no adversary can win the game with significantly greater probability than an adversary who must guess randomly. The most common indistinguishability definitions used in cryptography are:

- Indistinguishability under chosen plaintext attack (IND-CPA)
- Indistinguishability under (non-adaptive) chosen ciphertext attack (IND-CCA)
- Indistinguishability under adaptive chosen ciphertext attack (IND-CCA2)

Security under either of the latter definition implies security under the previous ones: a scheme which is IND-CCA secure is also IND-CPA secure, and a scheme which is IND-CCA2 secure is both IND-CCA and IND-CPA secure. Thus, IND-CCA2 is the strongest of these three definitions of security.

#### Non-malleability

Malleability is the property of some cryptographic algorithms. An encryption algorithm is said to be malleable if an adversary can transform a given ciphertext into another different ciphertext which decrypts to a related plaintext. That is given an encryption of a plaintext $m$, it is possible to generate another ciphertext which decrypts to $f(m)$, for a known function $f$, without necessarily knowing or learning $m$.

Malleability is not a desirable property for cryptosystems, since it enables an attacker to modify the contents of a message. For instance, suppose that a bank uses a stream cipher to hide its confidential information, i.e. financial transactions, and a user sends an encrypted message containing, say, "DEPOSIT £100 TO ACCOUNT 777." If an attacker is aware of banking network then he can easily intercept the message on the wire, and can guess the format of the plaintext message. Subsequently, the attacker could fabricate the message by changing the amount of the transaction, e.g. "DEPOSIT £10000 TO ACCOUNT 777." On the contrary, Non-malleability guarantees that given a ciphertext it is impossible to generate a different ciphertext so that the respective plaintexts are related.

## 2.8 Related Studies in Cryptographic Techniques

This section discusses the recent encryption schemes which are related to fog-enabled systems. More precisely, we discuss the schemes proposed for cloud assisted cyber physical systems (CCPS) and IoT.

### 2.8.1 Related Studies in Cloud-Assisted Cyber-Physical Systems

To secure communication in CCPS systems, the literature adopts a hybrid approach in which both symmetric (AES-based) and asymmetric (RSA-based and ABE) encryption techniques are used. AES is used to encrypt the communication between sensor nodes and the gateway, whereas the asymmetric schemes are used to encrypt the communication between the gateway and the service provider.

Kocabas et al. (Kocabas et al., 2016) present a general architecture consisting of acquisition, pre-processing, cloud and action layers for a medical CCPS system. They propose encryption schemes for secure data sharing at different layers. The study proposes the AES (Advanced Encryption Standard) symmetric key encryption scheme for communication between acquisition and pre-processing layers. The main disadvantage of (Kocabas et al., 2016) is the key management of symmetric keys in such a complex environment that is difficult to achieve.

Kim et.al (Kim et al., 2016b) propose an end-to-end message protection framework for CPS systems. The proposed scheme follows the publish-subscribe group communication model. The authentication servers are responsible for member authentication and key distribution. In the context of CPS, the devices are divided into a small number of groups, and end-point servers participate in all the groups. Each device is issued a long-term pre-shared symmetric key during the authentication phase and subsequently used for deriving the encryption keys from a random number sent per-message. The subscribers compute the decryption key when a message arrives.

Chen et al. (Chen et al., 2015b) propose a distributed authentication framework for the multi-domain M2M environment. The proposed framework applies a hybrid encryption scheme involving identity-based cryptography (IBC) and symmetric encryption with AES. It is assumed that the service provider and gateways are powerful enough to have high computation power to execute the IBC scheme, while the sensor nodes could only afford an efficient AES function for the encryption.

Hu et. al (Hu et al., 2016) propose a communication architecture for Body Area Networks (BANs) and design a scheme to secure the data communications between implanted/wearable sensors and the data sink/data consumers (doctors and nurses). They implement the CP-ABE and signature-based schemes to store the data in ciphertext format at the data sink (i.e. smartphone etc). The proposed scheme achieves role-based access control by employing an access control tree defined by the attributes of the data. This study makes an assumption that BAN devices should have certain computation capability to encrypt the patient's data and store the ciphertext into the data sink. However, most of the implanted devices have very limited storage and computation power and bilinear pairing based CP-ABE schemes are not suitable for them. The proposed scheme is based on the CP-ABE (Bethencourt et al., 2007) proposed by Bethencourt in which the size of ciphertext is proportional to the number of attributes in the access policy.

### 2.8.2 Related Studies in Constant-size ABE Schemes

Besides the above schemes, some studies propose constant-size ABE schemes for computationally limited lightweight devices. Compared to variable-size ABE schemes, constant size cipher-texts and constant size key based schemes incur less computational, communication and memory overhead.

Guo et al. (Guo et al., 2014) propose a CP-ABE scheme with constant-size decryption keys. Nuttapong et al. (Nuttapong Attrapadung, 2013) propose ciphertext-policy (CP-ABE) and key-policy (KP-ABE) schemes which support both monotonic and non-monotonic access structures with short ciphertexts.

Chen et al. (Chen et al., 2013) propose fully secure KP-ABE and CP-ABE with constant-size ciphertexts, and a fully secure ABS with constant-size signatures. The proposed schemes are based on inner product encryption/signature schemes. Oualha et al. (Oualha and Nguyen, 2016) extend the basic CP-ABE (Bethencourt et al., 2007) scheme using effective pre-computation techniques for bilinear pairing operations. Moreover, Li et al. (Li et al., 2017) present a CP-ABE system based on the ordered binary decision diagram (OBDD). The OBDD is a new method to define access structure and expressing access policies. Additionally, the studies (Chen et al., 2011), (Zhou et al., 2015), and (Emura et al., 2009) also propose constant-size ciphertext CP-ABE schemes.

### 2.8.3 Related Studies in Elliptic Curve Based Encryption Schemes

Recently, some studies propose elliptic curve based encryption and authentication schemes for IoT and CPS devices. Due to the lightweight security requirements of elliptic curves, such schemes are suitable for resource-limited devices.

Yaoa et. al (Yaoa et al., 2015b) proposed a lightweight no-pairing ABE scheme based on elliptic curve cryptography. The computational hard problems are based on Elliptic Curve Decisional Diffie-Hellman instead of bilinear Diffie-Hellman assumption. The proposed scheme is proved secure in attribute-based selective set model. The experimental results demonstrate the efficiency of the proposed scheme compared to bilinear-pairing based ABE schemes.

Odelu. et al (Odelu and Das, 2016) proposed a constant-size secret key CP-ABE scheme for lightweight devices. A major advantage of the proposed scheme is that it is based on elliptic curve cryptography. The secret key size is only 320 bits which is quite remarkable compared to existing CP-ABE schemes. However, the public key requires three points to represent each attribute in the attribute universe. This researcher believes that this scheme can further be improved and made more lightweight and efficient.

Mahmood et al. (Mahmood et al., 2018) propose an authentication scheme for smart gird. The major limitation of the proposed scheme is the scalability in case of large-scale CPS systems.

Challa et. al (Challa et al., 2017) present a signature-based authenticated key establishment scheme for IoT applications. The proposed scheme is based on elliptic curve cryptography (ECC).

### 2.8.4 Other Security Schemes

The existing security mechanisms, namely Secure Socket Layer (SSL), Transport Layer Security (TLS) and Kerberos, and various others schemes are proposed for homogeneous networks. These solutions lack scalability and may not be appropriate for heterogeneous CPS and IoT networks. Moreover, due to the high computational requirements of public key encryption, the SSL/TLS based approaches are also unsuitable. Considering the limitations of existing schemes, some studies propose adapted SSL/TLS based security schemes.

Kim et. al (Kim et al., 2017; Kim and Lee, 2017; Kim et al., 2016a) propose a secure network architecture consisting of local authorization entities for IoT. Compared to SSL/TLS mechanisms, symmetric keys are used for authentication and authorization. Any two entities namely client and server that wish to collaborate, first register themselves with the local authorization entity. *Auth* will subsequently generate the session key and distribute over a secure channel. For more details, interested readers are encouraged to read (Kim et al., 2016a). After receiving the session key, the ownership of session keys is verified. Upon successful verification, two entities securely communicate. Moreover, an open source secure swarm toolkit (Kim et al., 2017) is also proposed by the same authors. The proposed network architecture is quite promising and addresses the challenges of existing SSL/TLS based schemes. However, the management of session keys is a difficult challenge in large-scale IoT networks. Moreover, the compromise of local *Auth* entities can significantly impact the system.

## 2.9 Related Studies in Trust Models

In literature, trust models have been widely studied in several disciplines, namely Sociology, Psychology, Philosophy, Economics and Computer Science. The notion of trust is very complex such that no universally accepted consensus exists in the scientific literature. Generally, the concept of trust is tailored to fulfill the requirements of the underlying system. For instance, in computer systems, trust is related to security, privacy, reliability, capability, honesty and benevolence etc. Moreover, for dependable fog-enabled cyber physical systems, the notions of security, privacy and trust are highly related and critical issues.

This section discusses the recent trust management approaches for fog enabled cyber physical systems. But as fog computing is a new area of research so there are not many trust models. However, Fog-CPS systems share many commonalities with cloud computing, wireless sensor networks (WSNs), Internet of Things (IoT) and mobile ad hoc networks (MANETs). For these reasons, the trust models proposed for cloud computing, IoT and MANETs have also been considered.

Before discussing the existing works, a few of the recent surveys on IoT covering security, privacy and trust management issues are discussed. These surveys provide detailed insights on several dimensions of trust management and computation. Sicari et.al (Sicari et al., 2015) present a comprehensive review of studies focusing on several aspects of security, privacy and trust. However, in this survey the authors argue that satisfaction of trust requirements are highly correlated to identity management and access control. Maintaining this argument, all the cited trust studies are more or less related to securing distributed adhoc networks, user security, trust based access control, identity-based key agreement and node behaviour detection.

Guo et. al (Guo et al., 2017) present a survey of trust computation models for service management in IoT. The cited trust models are categorized on the basis of five design dimensions, namely trust composition, trust propagation, trust aggregation, trust update and trust formation. The survey is very comprehensive and gives several insights on existing trust computation techniques. Moreover, it also underlines the research gaps in existing models and further highlights the future research directions. Yan et. al (Yan et al., 2014) presents a literature review of trust management technologies for IoT. This survey considers several aspects such as trust context, and subjective and objective properties of trustee and trustor. It also underlines some objectives of trust management namely, 1) Trust relationship and decision, 2) Data perception trust, 3) Privacy preserving, 4) Data fusion and mining trust, 5) Data transmission and communication trust, 6) Quality of IoT service, 7) System security and robustness, 8) Generality, 9) Human-computer trust interaction and 10) Identity Trust. These objectives clearly embrace all trust management related facets in IoT systems.

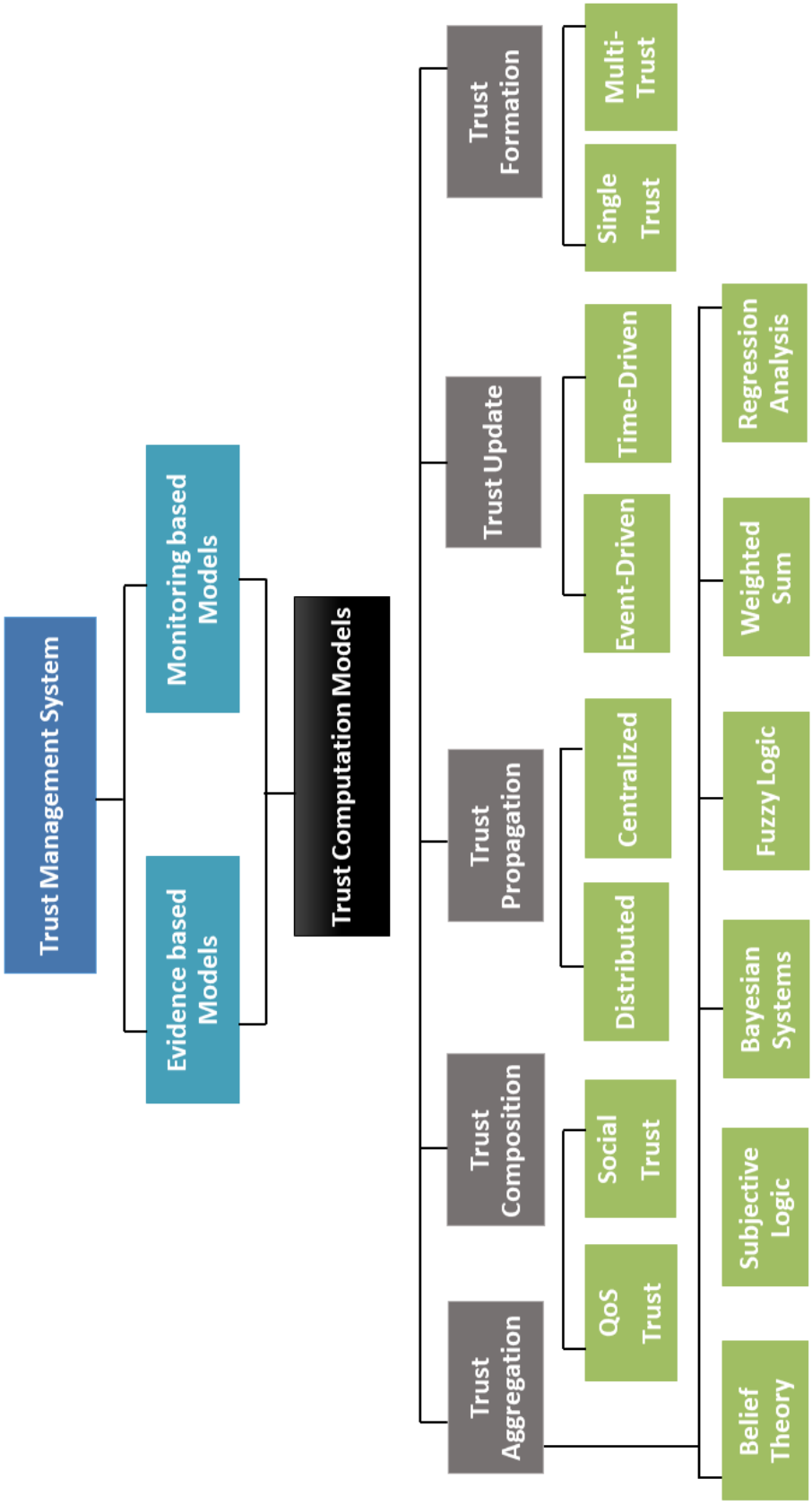


Fig. 2.6 Classification Tree of Trust Models

### 2.9.1 Classification of Trust Management Mechanisms

The classification of trust computation models presented in (Guo et al., 2017) is very detailed and comprehensive. However, it is not holistic as it does not take into consideration the two majors categories of trust management mechanisms. Taking this into account, the classification tree is further extended as shown in Figure 2.6. The trust management mechanisms can broadly be classified into evidence and monitoring based trust models. In evidence-based model, any information (i.e. public key, identity, contact number, email address, zip code etc.) that proves the identity of users is exploited to evaluate their trustworthiness. Public key cryptography is a good example of an evidence based trust model. In monitoring based trust models, the past interactions are observed to build the trustworthiness of entities.

For both these trust mechanisms, a number of trust computation models are proposed in the literature. Moreover, each of the trust models further incorporates several design dimensions, namely composition, propagation, aggregation, formation and update. Next, each of these dimensions is briefly discussed.

#### Trust Composition

Trust composition refers to different types of parameters, namely QoS and social trust which are considered in trust computation.

- **QoS Parameters** - Depending upon the application scenario, QoS is characterized by a number of different parameters. For instance, in cloud computing the QoS parameters include cloud resource attributes such as processing, storage, memory, and network capacity (Xiaoyong et al., 2015), (Sarbazi-Azad and Zomaya, 2014), (Kim et al., 2010) and (Rochwerger et al., 2009). In wireless networks, the QoS is quantified with packet forwarding ratio, energy consumption, and packet delivery ratio etc (Namal et al., 2015).

- **Social Trust** - Many recently proposed trust models for IoT systems are based on social trust. Under this social trust concept, IoT devices are considered more trustworthy if a relationship exists among the owners of IoT devices. The social trust is measured by friendship, honesty, community of interest, centrality, connectivity, and social contract (Nitti et al., 2014).

#### Trust Propagation

The dissemination of trust evidence to different entities of the system is referred to as trust propagation. There are two types of trust propagation schemes i.e. distributed and centralized.

- **Distributed** - In distributed schemes, the peers and/or nodes autonomously propagate trust evidence to other peers. Such schemes are particularly suitable for environments with distributed architectures such as peer to peer networks (P2P), mobile adhoc networks (MANET), and wireless sensor networks (WSN). Moreover, in distributed schemes, each entity maintains its own trust table.
- **Centralized** - In centralized propagation schemes, there is a dedicated node which maintains a trust table for storing the trust evidence of all nodes in a system. Any node can query the central node and gets the trust evidence. Centralized schemes are commonly used in cloud based systems.

### Trust Aggregation

After collecting the trust evidence, the next task is to aggregate it to quantify the trust. Major trust aggregation techniques investigated in the literature include subjective logic (Josang and Ismail, 2007) , weighted sum (Namal et al., 2015), belief theory (Ferrer et al., 2012), Bayesian inference (with belief discounting), multiple attribute decision making (MADM) (Xiaoyong et al., 2015) , fuzzy logic, logistic regression (Wang et al., 2014) , and regression analysis (Tian et al., 2017).

### Trust Formation

The trustworthiness of an entity can be evaluated based on a single or multiple parameters and/or dimensions. Precisely, there are two types of trust formation techniques i.e. single dimensional and multi-dimensional. Trust formation can be achieved by different techniques namely, weighted sum, minimum threshold, and trust scaled by confidence.

- **Single Dimension** - In this type of trust formation only one dimension of trust is considered i.e. service quality, honesty, and capability etc.
- **Multi-Dimension** - In multi-dimensional trust, several trust properties (i.e. competence, integrity, benevolence) are considered to evaluate the trustworthiness of an entity.

### Trust Update

Another important design aspect of trust computation model is trust update. In the literature, two types of trust update schemes, namely event-driven and time-driven are proposed.

- **Event-driven** - In this type of trust update, the trust tables are updated after every transaction or event. For instance, in service-oriented environments, a feedback is sent regarding the service quality to the trust manager in the cloud or recorded in the node itself.

- **Time-driven** - In the time-driven schemes, evidence (self-observations or recommendations) is collected periodically and trust is updated by applying a trust aggregation technique. In case if evidence is collected frequently, trust decay over time is applied to consider the recent trust evidence over past information. The exponential decay function with a parameter adjusting the rate of trust decay over time can be used depending upon the requirements of the specific applications.

Having presented the trust management mechanisms and the several aspects of trust computation, discussion now moves to the related work and indicates which of the above mentioned design dimensions are used in each study. Additionally, as mentioned above Fog-CPS systems share commonalities with cloud computing, MANETs, WSN, and IoT. The trust models related to these systems are discussed below.

### 2.9.2 Trust Models for Cloud Computing

As non-trivial extension of cloud computing, fog computing provides all compute, storage, and network services near the edge of the network. Fog nodes can be considered as small cloudlets. The authors in (Ni et al., 2018) underline that it is essential to evaluate the trustworthiness of fog nodes for dependable and secure fog-based IoT applications. Moreover, it is argued any trust management system proposed for fog computing should be decentralized, situation-aware, scalable and consistent. Following this, recent trust models proposed for cloud services are reviewed.

Majority of the studies compute trust on the basis of objective trust but some adopt a hybrid approach (Fan and Perros, 2014),(Ghosh et al., 2015),(Xiaoyong et al., 2015),(Nagarajan and Varadharajan, 2011) where trust is the fusion of objective and subjective evidence. The literature identifies two popular methods to compute objective trust, a) subjective logic (Josang and Ismail, 2007), and b) real-time adaptive trust evaluation approach (Xiaoyong et al., 2015), (Li et al., 2015). In adaptive trust evaluation approaches, the trust computation problem is modeled as a process of multi-attribute decision making (MADM) and weights are assigned adaptively either by maximizing deviation method (Xiaoyong et al., 2015) or information entropy (Li et al., 2015); whereas in subjective logic weights are assigned manually or subjectively (Josang and Ismail, 2007).

Nagarajan et. al (Nagarajan and Varadharajan, 2011) proposed a trust enhanced secure model (TESM) for trusted computing platforms. The authors argue that given the nature of both binary and property based attestation mechanisms, an attestation requester cannot be absolutely certain if an attesting platform will behave as per the expectation. TESM employs a hybrid trust model which is based on subjective logic to combine 'hard' trust from measurements and properties and 'soft' trust from past experiences and recommendations to reduce such uncertainties. However, the fusion of these trust factors, e.g. hard trust or soft trust, is not discussed in detail.

Habib et. al (Habib et al., 2013) proposed a trust-aware framework to verify the security controls considering consumers' requirements. The authors model the security controls in the form of trust properties. They subsequently introduce a taxonomy of these properties based on their semantics and identify the authorities who can validate the properties. The taxonomy of these properties is the basis of trust formalisation in their proposed framework. Equation 2.14 defines the trust model proposed in (Habib et al., 2013)

$$TM = (E, TR, OP), \tag{2.14}$$

where $E$ is the set of entities that share trust relationships, $TR$ is the set of trust relationships among the entities and $OP$ is the set of operations for management of the trust relationships. Furthermore, a decision model is proposed to assist consumers to choose trustworthy cloud providers. Moreover, trust evaluation is subjective and considers customers ratings for trust computation.

The study in (Ghosh et al., 2015) proposes a framework which combines trustworthiness and competence to estimate the risk of interaction. Trustworthiness is computed from personal experiences gained through direct interactions or from feedback related to reputations of vendors. Competence is computed based on the transparency of cloud service provider's SLA. Following this, the notion of general trust vector is discussed below. If a customer $c_j$ wishes to compute the trust of provider $p_k$ prior to the interaction, then a general trust vector model $GTV$ consists of seven-tuple as:

$$GTV = (P, A, I, \mathbb{D}, t, F, \mathscr{G}) \tag{2.15}$$

where $P$ is a set of providers, $A$ is a set of contexts over which previous interactions have occurred, $I$ is an interaction matrix of customer $c_j$, $\mathbb{D}$ is the trust domain, $\tau$ is the predefined temporal window, $F$ is a probability distribution function based on which the expected trust degrees will be assigned, and $\mathscr{G}$ is a function to evaluate the general trust vector. If a customer $c_j$ has previously interacted with the service provider $p_k$ over a predefined temporal

window $\tau$ then its history of interactions in considered in trustworthiness evaluation otherwise customer has to believe on the reputation of the said provider. Reputation is computed from the feedback of customers who have interacted with provider $p_k$. Trustworthiness of a service provider $p_k$ as perceived by a customer $c_j$ over temporal window $t$ is given as:

$$T^{\tau}(c_j, p_k) = \begin{cases} \mathscr{G}^{\tau}(c_j, p_k), & \text{If } C1 \text{ is true,} \\ \pi(c_j, p_k), & \text{Otherwise} \end{cases} \tag{2.16}$$

where $\mathscr{G}$ is a function to evaluate $GTV$ for provider $p_k$, $\pi$ is the overall reputation vector and $C1$ is a condition which returns true if $c_j$ has previously interacted with provider $p_k$ over a context similar or identical to current context $a_i$. $\mathscr{G}$, $\pi$ and $C_1$ are defined using Equations 2.17, 2.18 and 2.19:

$$\mathscr{G}^{\tau}(c_j, p_k) = \begin{cases} \frac{1}{|A|}\sum_{\alpha_i \in A} \quad \mu_{c_j}(p_k, \alpha_i), & \text{If } C1 \text{ is true,} \\ -\infty, & \text{Otherwise} \end{cases} \tag{2.17}$$

where $\mu$ is the expected degree of trust on provider $p_k$ during $t$ over context $a_i$.

$$\pi_{c_j}(p_k, \alpha_i) = \frac{1}{|\mathbb{D}|} \quad \sum_{i=1}^{|\mathbb{D}|} \mathbb{E}_{c_j}^{p_k}(d_i) \tag{2.18}$$

where $E$ is the state-based reputation. $\pi$ is based on Dempster-Shafer belief model.

$$C1 : |H_{c_j}^{\tau}(p_k, \alpha_i)| \neq 0 \tag{2.19}$$

Moreover, competence is assessed based on transparency in provider's service level agreement (SLA). Overall competence of a service provider $p_k$ in terms of any SLA $\phi$ is the mean of aggregated transparencies of all parameters is computed using Equation 2.20.

$$\mathscr{C}_{p_k,\phi} = \frac{1}{n}\sum_{i=1}^{n} \lambda_{param_i}(\phi) \tag{2.20}$$

where, $param_i$ is the $i$th parameter in the SLA $\phi$, $n$ is the total number of SLA parameters, $\lambda_{param_i}(\phi_i)$ is the transparency of parameter $param_i \in \phi$.

Lastly, the perceived interaction risk $\mathscr{R}$ in provider $p_k$ is modelled as:

$$\mathscr{R}(c_j, p_k) = k_1 \frac{1}{T^{\tau}(c_j, p_k)} + k_2 \frac{1}{\mathscr{C}(p_k)} \tag{2.21}$$

where $k_1$ and $k_2$ are the proportionality constants.

Ferrer et. al (Ferrer et al., 2012) propose the OPTIMIS trust model which is based on subjective logic and does not take into consideration user feedback. Li et. al (Li et al., 2014) propose a trust model for web services which considers the users' preferences and the impact of falsified ratings on trust evaluation. The proposed model is based on subjective logic and does not consider the real-time QoS attributes which make it impractical for cloud services.

Recently, Noor et.al (Noor et al., 2016) propose a reputation based trust management approach to compute subjective trust of cloud services. Trust is computed by taking feedback on multiple dimensions, namely usability, accessibility, availability, price and customer service etc. In this study, the trust result of a cloud service $s$ denoted as $T_r(s)$ is computed with Eq. (2.22).

$$T_r(s) = \frac{\sum_{c=1}^{|V(s)|} F(c,s) \times C_r(c,s,t_0,t)}{|V(s)|} \times (X \times C_t(s,t_0,t)) \tag{2.22}$$

where $V(s)$ denotes feedback given to the cloud service $s$ and $|V(s)|$ represents the total number of feedback. $F(c,s)$ are the feedback from $c$-th cloud service user weighted by the credibility aggregated weights $C_r(c,s,t_0,t)$. The credibility weight dilutes the influence of misleading feedback from attackers. $C_t(s,t_0,t)$ is the rate of change of subjective trust results. $X$ is the normalized weight factor for the rate of change of trust results which increase the adaptability of the model.

Additionally, a feedback credibility model is proposed to protect cloud services against malicious behaviors (e.g. collusion or Sybil attacks) from its users. A major limitation of this work (Noor et al., 2016) is that trust is computed from user feedback only. However, it is evident that both direct evidence and user feedback should be considered in assessing the trustworthiness of cloud services as the real-time evaluation of QoS parameters can give actual insights into the service quality.

### 2.9.3 Trust Models for IoT Systems

In this section, recent trust models proposed for IoT systems are reviewed.

Namal et. al (Namal et al., 2015) propose a trust management system for highly dynamic cloud-based IoT applications. The proposed model extends IBM 's MAPE-K feedback control loop. The autonomic trust management employs "Weighted Sum" for trust aggregation and considers multi-dimensional parameters, namely availability, reliability, capability and

response time for trust formation. Trust is computed using Equation 2.23.

$$T_{d,t} = \sum_{i=1}^{n} \beta_i(P_{d,t})i \tag{2.23}$$

where, $n$ is the total number of trust parameters, $P_{d,t}$ is a trust parameter for a IoT device $d$ at a time $t$ and $\beta_i$ is the weight assigned to the trust parameter. $P_{d,t}$ can be evaluated using Equation 2.24.

$$P_{d,t} = (\alpha P_{d,t-1} + (1-\alpha)C_{d,t})^{\frac{1}{b}} \tag{2.24}$$

where $C_{d,t}$ represents the current value for the trust obtained via transformation of the sensor raw data using Equation 2.25.

$$C_{d,t} = \big[\frac{-s(V_0 - V_{d,t})}{V_0 - V_{min}} r_1 + \frac{s(V_{d,t} - V_0)}{V_{max} - V_0} r_2\big] \tag{2.25}$$

In Equation 2.24, $\alpha$ is the weight given on the history which should be a value between 0 and 1. The value $b$ is a parameter that denotes by how much the calculated trust values are to be augmented or diminished.

A smart home environment is modelled in Matlab whereby availability, reliability, capability, and response time are quantified based on number of ping requests, bit error rate, the number of current sessions on a device and round-trip time respectively. However, a major limitation of this work is the inconsideration of security protection against attacks. IoT-based systems are at high risk of attacks, namely Sybil, collusion, ballot-stuffing and opportunistic service to name a few.

Nitti et. al (Nitti et al., 2014) propose a trustworthiness management system for social IoT. This work considered both QoS and Social trust parameters in trust composition. In order to evaluate the QoS provided by a node, service quality and computational capability parameters are considered. Likewise, for social trust, the centrality, relationship factors (ownership, co-location, co-work, social, and co-brand), and credibility of a node are considered. Trust is aggregated by static weighted sum. The overall trust degree of a service provider node is computed by centrality, objective, and subjective trust. Trustworthiness of a service provider node $p_j$ as seen by node $p_i$ is computed as follows:

$$T_{ij} = (1-\alpha-\beta)R_{ij} + \alpha O_{ij}^{dir} + \beta O_{ij}^{ind}. \tag{2.26}$$

Accordingly, $p_i$ computes the trustworthiness of its neighbours on the basis of their centrality $R_{ij}$, of its own direct trust $O_{ij}^{dir}$, and of the recommendations $O_{ij}^{ind}$ of the neighbours in

common with fog node $p_j$ ($K_{ij}$). All these addends are in the range [0, 1] and the weights are selected so that their sum is equal to 1. $T_{ij}$ is in the range [0, 1] as well.

Each node computes the trustworthiness of its friends on the basis of its own experience and on the opinion of the friends in common with the potential service providers. For more details about trust computation, interested readers are encouraged to read (Nitti et al., 2014). For trust update, the event-driven scheme is considered. The proposed trust model is appropriate for IoT systems.

Tian et. al (Tian et al., 2017) proposed a trust evaluation approach for sensor-cloud systems. Such systems are often susceptible to malicious attacks and can make sensor communication unreliable. In this work, the trust evaluation issue is formulated as a multiple linear regression (MLR) problem. Moreover, several parameters, 1) average energy consumption, 2) response time, 3) package delivery ratio, 4) maximum delivery distance, 5) position information and 6) list of communication objects, are considered for trust formation. However, only three parameters, namely average energy consumption, response time and package delivery ratio, are considered as features in MLR. Considering the energy restrictions of sensor devices, fog nodes are adopted to assist in the trust computation. Moreover, the least squares algorithm is used to find the fitting function between the communication features and the trust value.

### 2.9.4 Trust Models for MANETs

Wang et. al (Wang et al., 2014) proposed a logistic regression based trust Model for MANETs. The dynamic trust in a service-oriented MANET is modelled wherein a node can be both a service requester (SR) or a service provider (SP). In this work, trust formation is based on multi-dimensional parameters, namely energy-sensitivity, capability-limitation and profit awareness. These parameters are taken as features and each set of features has a corresponding trust value. The two classes of logistic regression classifier are trustworthy (0) and untrustworthy (1). The probability of trust being in one class or another is considered as the probabilistic statistical estimation of trust. Using the above parameters, the trust class at a recent time can be predicted on the basis of features at that time instance and historical evidence. The proposed model is evaluated on synthetically generated data using Poisson, Gaussian, and Normal distributions. Moreover, the trust in a service provider is evaluated on the basis of a service requester's own experience and recommendations of neighbouring nodes. More specifically, the trust evaluation is subjective. However, it is evident that the objective evidence is more reliable and enables an accurate trust estimation.

Shabut et. al (Shabut et al., 2015) proposed a recommendation based trust model for MANETs. For each node in the network, trust value $T_{ij}$ is calculated by combining both

direct and indirect trust values with different weights denoted by $w_d$ and $w_i$ respectively. $T_{ij}$ is computed according to Equation 2.27:

$$T_{ij} = w_d * T_{ij}^d + w_i * T_{ij}^i \tag{2.27}$$

where $w_d + w_i = 1$. The weights are used because of their significant impact on diminishing the possibility of wrong trustworthiness evaluation of direct and indirect trust information by nodes. Additionally, the proposed model includes a defence scheme which utilises a clustering technique to dynamically filter out attacks related to dishonest recommendations between certain times based on the number of interactions, compatibility of information and closeness between the nodes.

Li et. al (Li and Song, 2017) proposed a trust management scheme for vehicular ad hoc networks (VANETs). Dempster–Shafer theory and collaborative filtering techniques are used for trust aggregation. Trustworthiness of vehicles in VANETs is evaluated by data and node trust. Data trust is defined as the assessment of whether or not and to what extent the reported traffic data are trustworthy, and node trust is defined as how trustworthy the nodes in VANETs are. The node trust is further categorized into functional trust and recommendation trust.

Recently, Xia et. al (Xia et al., 2016) proposed a trust model based on Grey-Markov chain prediction technique to predict trust of nodes in MANET. The process of node trust assessment is based on node's historical behaviours, in which the trust decision factors include the subjective reputation and indirect reputation. The trust formation considers only one dimension i.e. packet forwarding.

Recently, Chen et. al (Chen and Wang, 2017) introduced the concept of social trust in VANETs referred to as vehicular social networks. Trust aggregation is achieved by Performance Evaluation Process Algebra (PEPA). Similar to most of the existing studies, trust formation is multi-dimensional and considers several parameters, namely 1) application domain, 2) friend entities, 3) neighbouring entities, 4) unknown entities, 5) history trust, 6) general trust and 7) vehicle profile. PEPA has superior features in compositionality and parsimony, which means that it can efficiently model systems with layered architectures and complex behaviours. PEPA also supports various numerical analyses through calculating its underlying continuous time Markov chains (CTMCs) directly or solving a set of approximated ordinary differential equations (ODEs).

Following the social trust concept, Wang et. al (Wang et al., 2018a) proposed a dynamic trust framework for opportunistic mobile social networks. A "two–hop feedback method" that requires intermediate nodes in a forwarding path to generate ACK messages to verify a node's honesty if they are two hops away is employed. Trust formation is multi-dimensional

based on parameters, namely 1) detection of behaviour, 2) delivery of trust, 3) processing of trust and 4) decision of trust.

### 2.9.5 Trust Models for Cyber-Physical Systems

In this section, the trust models proposed for cyber-physical systems are reviewed. Interestingly, for CPS systems, many different approaches, namely trusted computing, game theory and generic probabilistic graph modelling are employed.

Rein et. al (Rein et al., 2016) proposed the concept of trust establishment in a cooperative cyber physical system. The proposed model employs trusted event reporting to verify the authenticity of security related events in critical infrastructure. The correctness of monitored data and secret manipulation of monitoring equipment are achieved by a trusted information agent (TIA). The integrity of all system components is guaranteed by a chain of trust concept. A layered trust architecture whereby a trusted platform module (TPM) (i.e. dedicated security hardware chip) serves as a trust anchor and extends the trust to further system components. Each layer is responsible for computing the checksums of the components in the next upper layer. The proposed model is evaluated on a hydroelectric power plant and the overhead in data transmission between event source and data verification is analyzed.

The trusted computing technology, more specifically TPM, is exploited for trust establishment. A TPM is equipped with several cryptographic constructions, namely random number generation, remote attestation, binding and sealing. However, such computationally expensive operations are not suitable for resource limited CPS devices and IoT sensors.

Pawlick et. al (Pawlick and Zhu, 2017) adopt a game theoretical approach for trust computation in cyber physical systems. The proposed game of games modelling paradigm is called "strategic trust" and it captures the strategical and adversarial aspects of CPS security. The proposed framework consists of two simultaneous games, 1) FlipIt and 2) Signalling. In the FlipIt game, the attacker and defender attempt to control a common target i.e. cloud resource in this case. The signalling game models the decision of connected devices on whether to trust or not trust the commands received by the cloud.

The equilibrium outcome in the signaling game determines the incentives in the FlipIt game. In turn, the equilibrium outcome in the FlipIt game determines the prior probabilities in the signaling game. The Gestalt Nash equilibrium (GNE) characterizes the steady state of the overall macro-game. The novel contributions of this paper include proofs of the existence, uniqueness, and stability of the GNE. The proposed strategic trust is evaluated on a cloud-assisted insulin pump. The proposed game-theoretical approach is quite interesting and can be applied to detect and mitigate cyber threats in CPS.

Yan et. al (Wang, 2018a) adopted a perception-oriented approach to quantify trustworthiness in cyber physical systems. A multi-dimensional perception approach based on three major metrics of ability, benevolence, and integrity is considered. Ability measures one 's sensing and reasoning capability and influence to others. Benevolence captures the genuineness of intention and the extent of reciprocity in information exchange. Integrity provides the confidence about system dependability and predictability. These metrics are subsequently used for trust quantification.

The sensing, prediction, and communication functions of large-scale cyber-physical systems are modelled through a probabilistic graph model. The graph model provides a generic abstraction of scalable CPS networks. Moreover, the ability, benevolence, and integrity metrics are calculated based on the probabilistic graph model. Moreover, this work demonstrated that perception-level metrics can be calculated with the combination of Bayesian and statistical methods. Compared to other trust computation discussed above, a novel aspect of this work is the considerations of different CPS functions, namely sensing, prediction and communication, for trustworthiness quantification. The perception-based quantification method directly models subjectivity of beliefs and the influence of social behavior, with quantitative measures of ability, benevolence, and integrity, which have not been considered in other quantitative approaches.

### 2.9.6 Discussion on Trust Models

After reviewing several trust models and surveys, it can be concluded that trustworthiness of fog-enabled systems cannot be solely based on security and privacy aspects because an authenticated node can also act maliciously. Henceforth, it is essential to evaluate the trustworthiness of fog entities and CPS devices based on both security properties and performance indicators i.e. QoS parameters.

Moreover, it is observed that most of the existing studies on IoT systems, MANETs and CPS systems only assess the trustworthiness of sensors and CPS devices. However, there are other entities which are also vulnerable to cyber attacks and can be compromised. This is especially true for fog nodes which unlike cloud servers may be located in unprotected and hostile environments. Besides that, many trust models in IoT, MANETs and CPS systems also take into consideration the social aspects of trust. The ownership of devices and the relationship between different owners are exploited for trust computation. The social trust is essential in use cases whereby the interactions between humans and the devices owned by them are of primary concern. However, the main focus of this research is the interactions between autonomous devices in Fog-CPS systems. For these reasons, the social trust is not considered. Precisely, in the proposed TMS, a performance based trust computation approach

is employed. The trust of FOG-CPS entities is computed based on the QoS parameters and network communication features. The social trust might be considered in a future work.

Additionally, it is concluded that the trust computation is essentially a regression problem wherein the trust of an entity can be accurately predicted based on a set of features. For instance, the trust of a fog node can be estimated based on its computational and processing capabilities, response time, and task success ratio. Likewise, the trust of a CPS device can be based on the its performance and communication features. Regression analysis based schemes perform better than other trust computation models (i.e. subjective logic, weighted sum, and adaptive trust evaluation) as they consider several features to predict the trust. Such an approach is quite advantageous for a Fog-CPS system which is comprised of numerous entities with heterogeneous resources. One more advantage of employing the regression models for trust prediction is that in these models, the weights that best fit the input features are selected by regression over the entire training dataset.

There are several regression models namely, linear regression, multiple linear regression, logistic regression, support vector machine regression and random forest regression. Following this, the rationale behind employing random forest regression to predict the trust of Fog-CPS entities is discussed. The linear regression model is not employed to predict trust as it supports only one explanatory (independent variable) to predict the dependent variable. The logistic regression model predicts only two classes i.e. 0 and 1, however for proposed trust model a value between 0 and 1 was required. The multiple linear regression, support vector machine and random forest all take multiple independent variables and/or features to predict the dependent variable, trust in this case.

So, to choose the right regression model, experiments were carried out by employing multiple linear, support vector machine and random forest regression models. As the performance of a regression model is evaluated based on its accuracy calculated from mean square error (MSE). As per the expectation, random forest regression outperformed other models with 88% accuracy and 0.12 MSE. Ensemble models such as random forest are well suited for multidimensional data. The MSE in case of multiple linear regression and support vector machine were 0.15 and 0.18 respectively. Considering the prediction accuracy of random forest regression model, it is employed for trust prediction of fog nodes and CPS devices.

Another key challenge faced by the trust computation models of distributed systems is their vulnerability to malicious attacks. Similar to other distributed systems, namely P2P, MANETs and sensor clouds, the Fog-CPS systems are also vulnerable to self-promotion, bad-mouthing, ballot-stuffing, opportunistic service, on-off, Sybil, and collusion attacks. It is therefore essential to include countermeasures against these attacks.

The proposed trust management system is presented in the next chapter. The trustworthiness of both fog nodes and CPS devices is evaluated. The trustworthiness of fog nodes is evaluated by QoS service evidence gathered by Fog Assist (FA) node. The CPS devices also assess the fog nodes based on multi-dimensional communication parameters. Precisely, random forest regression is employed for trust computation. The regression based trust evaluation approaches are better than subjective logic as they do not require weight inputs from experts. Additionally, as mentioned above all Fog-CPS entities namely, fog nodes, FA nodes and CPS devices are also vulnerable to malicious attacks. The compromised entities can fabricate the evidence to change the trust of other entities. The identification of malicious parameters and the rate of change of trust are vital to detect data anomalies, Sybil, collusion and other attacks, and subsequently computing a precise and accurate trust of fog nodes. Considering the limitations of existing studies, a new trust management system is proposed in which fog nodes and CPS devices are evaluated on a multi-dimensional criteria. Moreover, a trust credibility evaluation model is designed to adjust the trust of Fog-CPS entities in three cases whereby the CPS devices, fog nodes and FA could be compromised.

## 2.10 Summary of the Chapter

This chapter has thrown light on various aspects related to Fog-CPS systems. Section 2.1 discussed the fog computing architectures, its features and the standards. It further reviewed related studies in a bid to identify their limitations.

The integrated frameworks were discussed in Section 2.2. The security schemes were reviewed in Section 2.3. The background of elliptic curve cryptography is discussed in Section 2.4. The fundamentals of pairing based cryptography and pairing schemes were discussed in Sections 2.5 and 2.6 respectively. Functional encryption schemes were briefly discussed in Section 2.6.1. Additionally, the security evaluation techniques were outlined in Section 2.7. The related security schemes particularly cryptographic techniques were discussed in Section 2.8. Moreover, the related TMS and trust models were discussed in Section 2.9.

# Chapter 3

# A Secure Integrated Framework

This chapter presents the proposed secure integrated framework. Firstly, it presents a general architecture of the fog-enabled smart grid power control systems (Fog-SGC), their security and trust challenges. It further states the security requirements of the Fog-CPS systems. Following that, it explores how the proposed framework can address the security and trust challenges faced by the Fog-CPS systems. As the framework consists of two components, i.e. SC and TMS. The discussion first focuses on how the SC can achieve the security goals, namely ***data confidentiality***, ***authentication*** and ***authorization***. Following that, a lightweight encryption scheme which is proposed as part of the SC is presented. Moreover, a TMS, the second component of the proposed framework, is presented. The proposed TMS addresses the trust challenges faced by the Fog-CPS systems.

## 3.1 Fog-enabled Smart Power Grid Control System

In this section, the proposed fog-enabled smart grid power control (Fog-SGC) application scenario is presented. Further, the security, privacy and trust challenges inherent to such an environment are discussed. Following these, the security and trust requirements of Fog-SGC systems are also outlined.

Fog computing can improve the monitoring and management of next generation smart grids. The pervasive CPS devices i.e. internet connected smart meters and smart appliances can support the automation of future distribution grids. A fog-enabled smart grid power control (Fog-SGC) is advantageous in several ways, for example for:

- enabling efficient electricity transmission,
- timely restoration of electricity after power disruptions and/or outages

- reducing the electricity bills, management and operational costs
- effective integration of renewable energy sources to existing power grids in order to decrease reliance on fossil fuels and reduce CO2 emissions,
- increasing the dependability of power grids by achieving reliability, availability and efficiency.

**Novelty**

Generally, the fog computing does not follow a single architecture model. In many cases, the design of a fog deployment model depends upon the specific use case under consideration. Large-scale smart grid power control services are pervasive in cyber physical environments consisting of millions of smart meters, sensors and compute resources. However, an effective orchestration service is required for the efficient management of fog services, and tackling their dynamic variations and on-demand operational behaviours. The orchestration of fog services and resources would enable robust maintenance and enhancement by achieving reliability, dependability and security. In order to tackle these challenges, the basic fog computing deployment model is extended and dedicated fog nodes called "Fog Assist" are added in the *fog layer*. The Fog Assist (FA) nodes could be tasked with service matching, provisioning and trust management (Wen et al., 2017).

### 3.1.1 The Proposed Deployment Model

The Fog-SGC as shown in Figure 3.1 consists of three layers, namely *CPS devices*, *fog communication* and *cloud*. In the *CPS devices* layer, there are smart home devices, smart building devices and smart factory devices in customer premises. The CPS devices such as smart meters interconnect the appliances and subsequently report the power consumption of the premises. The power generation and distribution centres are also part of this layer.

The power grids generally cover large geographical areas, thus to monitor the power consumption patterns and to manage the supply and demand requirements, different devices, namely Neighborhood Area Networks (NANs) and Wide Area Networks (WANs) are added to the *fog communication* layer. The NAN devices monitor small geographic areas and forward the power readings of neighbouring smart meters to distribution access point (DAP). The DAP further aggregates them and generates summaries of power consumption. The fog nodes with high computation powers serve as Wide Area Networks (WANs) and interconnect multiple NAN devices. WAN devices also act as Remote Terminal Units (RTUs) and transmit meter data to the *cloud* layer.

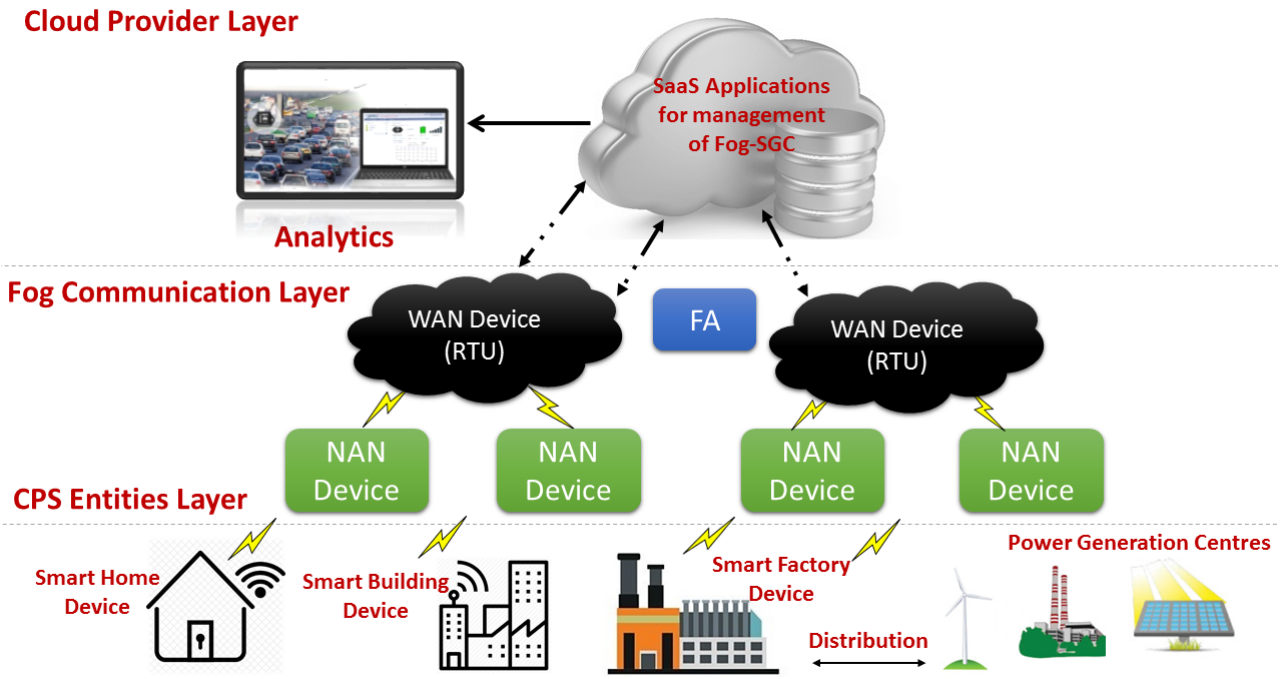


Fig. 3.1 Fog-Enabled Smart Power Grid Control

Moreover, there are "Fog Assist" (FA) nodes which are dedicated for entity registration, service orchestration, service provisioning and trust management. Lastly, the *cloud* layer hosts SaaS applications for the acquisition, processing, presentation and management of the metering data. Furthermore, there are advanced electricity monitoring systems (AEMS) which are powered by analytics on real-time metering, demand and supply data to forecast power consumption patterns.

### 3.1.2 Attacks on Cyber Physical Systems

Despite the opportunities provided by the Fog-CPS systems, they face increased security, privacy and trust challenges. CPS systems bring significant challenges when devices lack security protection. Recent cyber attacks on CPS systems including Ukrainian power grid (Lee et al., 2016), DNS provider Dyn (Lewis, 2017) and others underline the threat of connectivity and the vulnerability of resource constrained devices to be compromised. Following this, each of these attacks is briefly discussed.

1. **Stuxnet Worm:** One of the very well known attacks on CPS systems is the Stuxnet worm (Kushner, 2013) on Iranian nuclear facilities. It stealthily spread between computers running Windows by exploiting the ***privilege escalation*** vulnerabilities.

2. **Power Grid:** The blackout of power grid (Lee et al., 2016) in Ivano-Frankivsh region of Ukraine is another key attack which demonstrates how the resource constrained devices could be compromised and bring the CPS systems to halt. In this incident, the attackers successfully gained access to the Supervisory Control and Data Acquisition (SCADA) system of the grid, and subsequently caused the power outage for several hours.

3. **Dyn Internet Service Provider:** In the case of Dyn, the distributed denial of service (DDoS) attack was launched by small compromised devices which formed a botnet and collapsed the fundamental infrastructure comprising the Internet. In this attack, several social media (i.e. Twitter, Facebook) and other (i.e. PayPal, Amazon and Netflix etc) websites were down and/or inaccessible. It is later found that in Dyn attack, small computing devices such as printers, webcams, residential gateways and baby monitors were exploited to form a bot network. The Dyn attack was a reminder that IoT botnets could be a huge threat to the future internet services.

4. **Traffic Light Control:** Additionally, a traffic controller in streets of Ann Arbor, Michigan was compromised to manipulate the traffic lights by Ghena et. al (Ghena et al., 2014). Direct radio communication with the traffic controller was exploited to send fabricated signals to traffic lights.

5. **Autonomous Vehicles:** Along with other CPS systems, the autonomous vehicles are also vulnerable to cyber attackers. One such attack (Andy, 2015) has recently been carried out by Sammy Kamkar who had invented a device costing only $100. The device was used to attack General Motor's OnStar communication systems, and allowed an attacker to track, unlock and start the vehicle.

6. **Illegal Network Access:** Public networks are always vulnerable to illegal network access. In 2014, Kaspersky Lab found that more than 1 million devices are affected by 3.5 million malware. The malware were stealing the user credentials in a bid to gain access to the services and host networks.

### 3.1.3 Security, Privacy and Trust Threats and Challenges in Fog-CPS Systems

The security and privacy challenges faced by cloud computing are not fully addressed yet. In past few years, several security incidents (e.g. Sage, Sony Pictures, Home Depot, Target, iCloud Hack) have been reported. Moreover, the global tech giants and cloud vendors

such as Google, Amazon, and Yahoo have also suffered from security and data breaches. Cloud security has become an important factor hampering the adoption of cloud computing. However, owing to the following characteristics features of fog computing, it is deemed to a more secure architecture than cloud computing:

- The fog nodes located near the network edge transiently process and analyze the data reported by CPS devices. With this approach, data is not sent to the cloud servers, which subsequently decreases dependency on the Internet network. Similar to in-house data centres and private clouds, the data storage and processing at local fog nodes is more secure. Hackers cannot easily gain access to private data.

- In fog-based systems, the data transmitted by the CPS devices is processed and analysed on local fog nodes. With the non-real-time data transmission between Fog-CPS entities, eavesdropping is not as useful as in the case of cloud-based cyber physical systems where all the sensory data is sent to the cloud for further processing and subsequently affecting the physical environment in real-time. However, the local fog nodes can still process and/or gather the sensitive information of users, if proper privacy preserving methods are not employed.

On one hand, fog computing overcomes a few of the weaknesses of the cloud computing. But, on the other hand, it also inherits the existing security, privacy and trust challenges. However, fog computing has some specific security, privacy and trust threats due to its unique characteristics such as low latency, decentralized infrastructure, location awareness and mobility support. The security, privacy and trust challenges faced by the Fog-SGC systems are discussed next.

In Fog-SGC systems, machine-to-machine (M2M) communication can take place between autonomous embedded devices, sensor nodes, actuators, fog nodes and cloud services. M2M communication can be compromised via several attacks namely:

- Interception
- Interruption
- Modification
- Fabrication
- Unauthorized Authentication and Authorization

These principal attack mechanisms further break down to following security, privacy and trust challenges. A malicious attacker may launch them to disrupt the fog computing. Some

of the following attacks are identified by (Ni et al., 2018), but they are discussed as the Fog-SGC systems also face them.

1. **Data and Usage Privacy:** The fog nodes are located near the end devices and are therefore more aware of surrounding environment. They can gather information about all the connected CPS devices. Privacy is a critical issue in Fog-SGC systems as the sensitive data is often shared during collection, processing and transmission. Data owners are not willing to share their private information with others, but the leakage of sensitive data cannot be avoided. The privacy includes four aspects, that is, identity privacy, data privacy, usage privacy and location privacy. Additionally, smart meter readings pose a high risk to the user privacy. More specifically, as reported by many existing works (Li et al., 2014) (Yassine et al., 2015), all activities of daily living (ADL) can easily be inferred from smart meter readings.

2. **Unauthorized Secondary Usage:** Power consumption readings and user data could be exposed to an untrusted party, whilst they are being processed on fog nodes, and transmitted between two entities.

3. **Data Correlation:** The data stored and processed by fog nodes can easily be correlated to generate a device profile which can subsequently be used for marketing and advertising purposes. Additionally as reported by (Yassine et al., 2015), the smart meter readings can reveal a range of information about consumers, such as how many people are in the home, the types of appliances they use, their eating and sleeping routines, and even the TV programs they watch.

4. **Identity Impersonation:** A malicious attacker could pretend to be a legitimate user/device to provision services from fog nodes. It could also impersonate a legitimate fog node to offer low quality or unreliable services to CPS devices.

5. **Sybil Attack:** Robust identity and access management is a pre-requisite for the Fog-CPS systems, more specifically the Fog-SGC systems. Without such mechanisms in place, Sybil attackers could easily generate the fake identities and pseudonyms for CPS devices and fog nodes. The attackers can later use them to compromise or control the Fog-SGC systems. Moreover, the private information of users can also be used for other malicious purposes, namely phishing, spamming and targeted advertisements etc.

6. **Collusion Attack:** In Collusion attacks, several parties collude to achieve some common goal and/or gain an unfair advantage. In Fog-CPS systems, the CPS devices

can collude to manipulate the trust values of fog nodes. The colluding entities aim to affect the accuracy of the TMS, more specifically the trust computation process. Additionally, in fog computing, fog nodes and CPS devices may collude to increase their attack capability and subsequently attack the Fog-CPS system.

7. **Privilege Escalation:** In this type of attack, an adversary can escalate the privileges of a user in order to give him/her access to confidential data and/or to a higher role. The user in the escalated role can further carry out actions to disrupt the operation of Fog-SGC systems.

8. **Identity and Access Management:** Large scale Fog-SGC systems are comprised of several thousand CPS devices and processes. Robust identity and access management (IAM) is the pre-requisite to ensure security in such systems. IAM enables the individuals to access the corresponding resources. IAM also prevents the identity impersonation, unauthorized access and data disclosure attacks.

9. **Forgery:** In Fog-SGC systems, the malicious attackers may forge the identities of CPS devices and fog nodes. The forged identities can be used for several malicious purposes including disseminating fake information, and reporting false meter readings and communication parameters which are subsequently used in trust computation. The forged information or resources may also likely to consume excessive network resources such as bandwidth and energy to name a few.

10. **Spam:** In spamming attack, the malicious attackers generate bogus content and redundant information, and subsequently broadcast them to all other nodes in a network.

11. **Eavesdropping / Man-in-the-Middle Attack:** In Man-in-the-Middle (MITM) attack, a malicious attacker secretly listens over the communication channel. With regard to Fog-SGC systems, the eavesdropper can capture packets being transmitted between CPS devices fog nodes. It can subsequently modify them and relay back the fabricated data. However, the two parties would still believe that they are directly communicating with each other. MITM attack can easily be tackled by encrypting the data.

12. **Denial-of-Service:** In DoS attacks, the fog nodes are flooded with superfluous requests such that legitimate CPS devices and other entities could not provision services from them.

13. **Tampering:** In this type of attack, the compromised devices purposefully drop, delay or modify transmitting data, with the aim of attacker being able to disrupt a Fog-SGC system and degrade its efficiency.

14. **Physical Damage and Node Capture:** Due to operation in open environments, CPS devices and fog nodes are under great risk of physical damage and node capture. Furthermore, the CPS devices have limited resources and could therefore be easily compromised and broken.

15. **Jamming:** An attacker deliberately generates a huge number of bogus messages to jam communication channels or computing resources, such that legitimate users and/or CPS devices are prohibited from normal communication and computation.

### 3.1.4 Security Properties Violation

The above mentioned attack mechanisms lead to the violation of following security properties.

1. Confidentiality
2. Integrity
3. Availability
4. Privacy
5. Authentication and Authorization

### 3.1.5 Key Challenges in Designing Secure and Robust Solutions for Fog-CPS Systems

As recognized by many researchers including (Singh et al., 2016), the main challenges in the security of the cyber-physical systems (CPS) include:

1. **Heterogeneity:** All Fog-CPS systems including smart power grid, autonomous vehicles and smart farming have different resources and security requirements. Some require very high computational resources and robust security mechanisms, while others do not. For example, safety-critical systems such as drones and self-driving cars require the robust and efficient authentication and authorization solutions. However, for payment systems such as "Samsung Pay" and "Apple Pay" high performance is as equally desirable as the confidentiality and authentication of transactions. Similarly, in case of battery-powered devices, the lifetime and availability are considered just as important as the data security. Moreover, for some sensor applications, ensuring data integrity is more crucial than confidentiality. Precisely, all Fog-CPS systems and CPS devices have different security requirements and therefore emphasizing on one solution

that fits all is difficult to propose. However, one way to possibly address this challenge is to devise lightweight, innovative and robust security solutions which can be used across numerous heterogeneous devices. Elliptic curve cryptography is lightweight but secure and can therefore be employed to secure Fog-CPS systems.

2. **Operation in Open Environments:** Unlike cloud based systems, the Fog-CPS systems might operate in open and hostile environments. This opens up an entirely new class of attacks that Fog-CPS systems might face including but not limited to, illegitimate access through mediums other than traditional networks (e.g., physical access, Bluetooth and radios). The threat model of cloud-based systems is well-understood and several protection mechanisms have already been designed to counter the potential threats and attacks. However, the Fog-CPS systems also require robust security solutions which should be designed by considering this new threat model.

3. **Scalability:** Lastly, another key challenge faced by Fog-CPS systems is their unprecedented scalability. Precisely, according to Cisco, 50 billion devices will be connected to the Internet by 2020 (Evans, 2011) and this number will reach 500 billion by 2025 (Camhi, 2015), far exceeding the world population. However, any security solution proposed for such systems must scale accordingly, to be more specific, the overhead of adding and removing devices to/from the security solution should be minimal.

### 3.1.6 Security Requirements of Fog-CPS Systems

1. **Authentication:** Fog-CPS entities, namely CPS devices, fog nodes and cloud services operating in different realms offer and provision services in real-time. As a result, such systems face numerous security, privacy and trust challenges emerging from both cyber and physical spaces. Moreover, in such cases, ensuring trustworthy behaviour of Fog-CPS entities is non-trivial. One way to solve this could be through robust and efficient authentication and authorization mechanisms. It is essential that each entity should have a unique identity and should only access the resources, it is granted permission for. Without appropriate security mechanisms, any malicious attacker can compromise the resource constrained devices and target the fog services in ways elaborated in Section 3.1.3. Moreover, the attackers can also take advantage of lack of authentication schemes and get away without leaving behind any trace of malicious activities. Precisely, illegitimate access to Fog-CPS resources can only be prevented through robust identity and access management systems.

2. **Authorization:** Apart from authentication, a robust authorization system is indispensable for Fog-CPS systems. The lack of access control mechanisms enable the malicious attackers to harm the Fog-CPS systems in several ways such as, privilege escalation, identity impersonation and unauthorized access. In Fog-CPS systems, the access control policies are defined by fog nodes to enforce authorization mechanisms in every trust realm of these systems. Similar to ABE schemes, the access control policies must take into consideration the attributes (e.g. IP address, MAC address, trustworthiness, geographical location and resource ownership etc) belonging to Fog-CPS entities, as they can establish their unique identities.

3. **Lightweight Security Solutions:** The CPS devices have low computational capability and therefore cannot execute computationally complex and resource intensive operations. The security solutions designed for CPS devices should be very lightweight and fast. The lightweight cryptographic solutions, namely elliptic curves, blockciphers, hash functions, streamciphers and one-pass authentication, could be appropriate for resource constrained devices.

4. **Resilience to Sybil Attacks:** Fog computing is vulnerable to Sybil attacks, in which attackers are able to manipulate faked identities and abuse pseudonyms to compromise real-time services and CPS applications (Ni et al., 2018). In the presence of Sybil attackers, the normal CPS devices may be misled by the faked data and the CPS applications may generate incorrect results. A Sybil attacker may broadcast spam and advertisements or disseminate malware and phishing websites to steal the private information of users.

   Unfortunately, most Sybil attackers behave similarly to normal users, how to detect the presence of Sybil attackers and thereby identify the Sybil attackers is extremely difficult, which makes Sybil defense of paramount importance. To detect Sybil attackers, the basic information of normal users is needed for the detector to compare the difference between normal users and Sybil attackers, such as social graph, social community, behavior pattern and friend relationship.

5. **Trust Management:** Security solutions such as identity and access management (IAM), and encryption can secure the communication between collaborating entities and enforce robust authentication and authorization. However, these mechanisms do not guarantee that all entities are fully trusted. Trust plays an important role in establishing dependable relationships among Fog-CPS entities. Furthermore, in Fog-CPS systems, the CPS devices can provision low-latency services from fog nodes in the vicinity. Likewise, the fog nodes can also cooperate with other fog nodes to

provide real-time services. Nevertheless, some fog nodes may not maintain quality of service (QoS) due to various factors such as service cost, energy usage, application characteristics, data flow, and network status. Precisely, before any prior interaction, both CPS devices and fog nodes have no idea about how their partners would behave. Therefore, selecting fog nodes with high trustworthiness to cooperate with is quite important in the implementation of CPS applications and services. Because without trust mechanisms, such interactions are subject to risk and uncertainty that an entity might experience.

### 3.1.7 Discussion

In the preceding sections, the security, privacy and trust challenges faced by Fog-CPS systems are underlined. The difficulties in designing secure solutions are also outlined. Following that, the security requirements of Fog-CPS systems are highlighted. It is noted that these aspects cover an all encompassing view of the Fog-CPS systems. However, this dissertation focuses on solving the security and trust challenges. The security challenges include ***data confidentiality***, ***authentication*** and ***authorization***.

## 3.2 The Proposed Secure Integrated Framework

The Fog-CPS systems bring together inherently open, distributed and heterogeneous resources. Due to the open and distributed nature of Fog-CPS systems, they face several security, privacy and trust challenges. In Fog-CPS systems, inter-device and inter-system collaborations are subject to risk and uncertainty. CPS devices have become a new weapon for hackers to break into these systems via cyber and physical spaces. CPS devices are vulnerable to cybersecurity challenges and threats due to their resource limited capabilities and lack of protection mechanisms.

To be more specific, Fog-CPS systems face numerous security threats, namely eavesdropping, packet payload manipulation, DoS and man-in-the-middle attacks etc. The communication between CPS devices, fog nodes and cloud service providers (CSP) can be compromised by any of these attacks or a combination of them. These attacks subsequently lead to the lack of ***data confidentiality***, ***authentication*** and ***authorization***.

On top of that, in Fog-CPS systems, ***data confidentiality*** measures are required to ensure no unauthorized CPS devices, fog nodes and/or malicious attackers can gain access to the sensitive data related to the system. For instance, if proper security mechanisms are not employed in Fog-SGC systems, electricity consumption readings could easily be modified

and false meter readings could be reported. Likewise, with the lack of ***authentication*** mechanisms, any malicious device and/or system can easily join/leave the Fog-CPS network. It can also get away without being caught even if it performs maliciously. Additionally, a rogue node can impersonate a legitimate fog node and trick the CPS devices to use its services. It can do harm to the system in several ways as listed below:

- providing low quality of service
- gathering sensitive information from connected CPS devices and/or fog nodes
- attempting to escalate its privileges to get access to other resources
- compromising the vulnerable CPS devices and fog nodes and subsequently using them as malware bots to form a botnet.

Besides authentication, another key aspect related to security is the ***authorization***. The authorization mechanisms define who can access what resources in a given system. They also implement policies to enforce selective restriction to confidential information, data and resources. Similar to other open distributed systems, the Fog-CPS systems also require robust authentication and authorization techniques the absence of which can cause havoc, as is evident from the recent attacks on CPS systems (Kushner, 2013), (Lee et al., 2016), (Lewis, 2017) and cloud-based (Lewis, 2014), (BBC News, 2016), (BBC News, 2014) systems. Precisely, the lack of robust security techniques can disrupt the operation of Fog-CPS systems and subsequently affect their dependability and availability.

Apart from security challenges, Fog-CPS systems also face numerous trustworthiness issues. The decentralized architecture and mobility support features of fog computing introduce trust challenges in Fog-CPS systems as listed below:

- Fog computing does not necessarily follow a pre-defined network architecture. The fog nodes can be added and removed depending upon resource requirements of a specific use case under consideration. Some fog nodes which were part of the network could no longer exist and need to be replaced with other fog nodes. This could happen for various reasons including, the inability to provide quality of service (QoS), load balancing, service cost, energy usage, node capture or some other attacks. The inability to provide QoS raises concerns about the dependability and reliability of Fog-CPS systems.
- In the events of node mobility, data and processes running on fog nodes need to decoupled properly such that they cannot be modified and/or misused by malicious parties.

- Additionally, similar to CPS devices, fog nodes are also deployed in open and untrusted environments and are therefore at the risk of being compromised, broken and stolen.

Precisely, before any prior interaction, both CPS devices and fog nodes have no idea about how their partners will behave. Without trust mechanisms, such interactions are subject to risk and uncertainty that an entity might experience.

Having discussed the security and trust challenges, it is clear that establishing trustworthy and dependable Fog-CPS systems requires a multi-dimensional and multi-faceted approach which considers both the security and trust issues. The protection of Fog-CPS systems demands comprehensive vulnerability analysis, and extensive theoretical and practical innovations in security and trust technologies. Besides that the proposed solutions must be lightweight such that they could be applied in resource limited CPS devices. Considering the limitations of existing works (elaborated in Chapter 2), this dissertation proposes a secure integrated framework for Fog-CPS systems.

As discussed in Chapter 1, the framework consists of two components, namely 1) a security component (SC) and 2) trust management system (TMS) as shown in Fig. 3.2. The

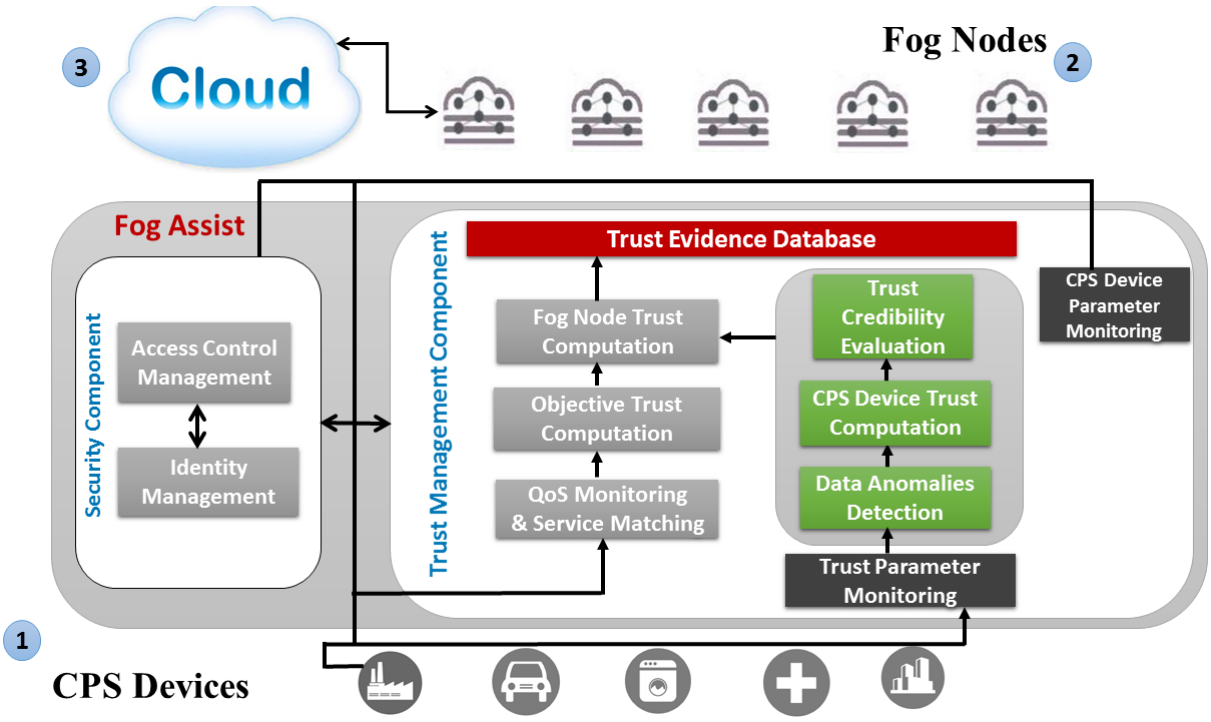


Fig. 3.2 Integrated Framework for Fog-CPS Systems

SC component establishes identity and access control in Fog-CPS systems. TMS evaluates the trustworthiness of Fog-CPS systems by computing trust for each of the entities, namely fog nodes and CPS devices.

### 3.2.1 Security Component

This researcher believes the security goals i.e. ***data confidentiality***, ***authentication*** and ***authorization*** can be achieved by employing cryptographic techniques. These goals can further be categorized as an identity and access management problem. So considering these different aspects, the first component of the proposed secure integrated framework was designed. The SC component consists of two subcomponents, namely *Identity Management* and *Access Control Management*. The different subcomponents of the SC enable FA nodes to efficiently manage the *identity* and *access control* in Fog-CPS systems. It implements the access privileges which are defined by fog nodes and makes sure that individual CPS devices and/or users are granted (or denied) those privileges under appropriate circumstances and/or network conditions. Access Policies also define how the sensory data is utilized in other services offered by the fog nodes.

#### Identity Management

In Fog-CPS systems, it is essential that all devices, sensors, monitors and fog nodes must have trusted and unique identities. Trusted identities would guarantee that all CPS devices and fog nodes are authenticated. Inline with this approach, the concept of unique attributes is introduced to avoid impersonation and Sybil attacks in open and distributed Fog-CPS systems. The set of unique attributes in the proposed Fog-CPS scheme can prevent impersonation, forging identities and Sybil attacks. Precisely, a device needs a set of unique attributes to successfully register with the FA node. The attributes can further be employed in ABE schemes to generate the secret keys and subsequently encrypt the information.

However, it is underlined that unique attributes cannot prevent the malicious behaviour and therefore thwart masquerade attacks. An authenticated node can also act malicious. However, it is underlined that the proposed secure integrated framework is designed to countermeasure the malicious activities of even the authenticated nodes. To be specific, the TMS can evaluate the trustworthiness of all Fog-CPS entities and based on which computes its trust value which is subsequently used to define their access control rights. The TMS acts supportively when an authenticated CPS device act maliciously.

#### Access Control Management

In Fog-CPS systems, access control management is required to determine which entity (a device or a user) can access what resources, for example, read or write data, execute programs and control actuators. However, due to the resource constrained nature of CPS devices, they lacked proper mechanisms for access control. Similar to cloud services, the fog nodes define

access policies for CPS devices and other nodes. Any device which fulfills the access policy can request and/or provision resources. Moreover, in the proposed framework, FA first authenticates the Fog-CPS systems and then implements the access control policies defined by the fog nodes. In the proposed framework, the objectives of secure identity management, authentication and access management are achieved by employing lightweight cryptographic techniques. To be more specific, as a part of the SC, a novel lightweight encryption scheme based on elliptic curves is proposed to enforce robust authentication and access control in Fog-CPS systems. The encryption scheme is presented in Section 3.3.

### 3.2.2 Trust Management System

The TMS of the proposed secure integrated framework consists of nine subcomponents:

1. QoS Monitoring and Service Matching
2. Objective Trust Computation
3. Trust Parameter Monitoring
4. CPS Device Parameter Monitoring
5. Data Anomalies Detection
6. CPS Device Trust Computation
7. Trust Credibility Evaluation
8. Fog Node Trust Computation
9. Trust Evidence Database

The different components of TMS enable FA to accurately and precisely compute the trust of fog nodes and CPS devices respectively. For fog nodes, trust is computed by aggregating the QoS evidence monitored by FA, and the network communication parameters, namely energy consumption, bandwidth and response time reported by the CPS devices. Likewise, the trust of CPS devices is computed based on communication features monitored by the fog nodes. A ***Trust Parameter Monitoring*** module is installed in each CPS device and fog node. It quantifies the latency, energy consumption and bandwidth utilized in communication between the fog nodes and CPS devices, and vice versa.

***QoS Monitoring and Service Matching*** module assists FA in evaluating the service quality and finding the services matching the user requirements. The QoS evidence is later

fed into the ***Objective Trust Computation*** module in order to compute the objective trust. FA subsequently stores all the monitored parameters and trust results in the ***Trust Evidence Database***.

Moreover, any anomalies in parameters monitored and/or quantified by fog nodes and CPS devices are detected by ***Data Anomalies Detection*** component prior to being incorporated into ***CPS Device Trust Computation***. Next, the ***Trust Credibility Evaluation*** module finds the discrepancies in trust computed in consecutive time instants. The trust inconsistencies are analyzed to prevent the malicious behaviour of Fog-CPS entities. Lastly, the objective and CPS trust are sent to the ***Fog Node Trust Computation*** module to compute the final trust of fog nodes. The details of each of these modules are given in Section 3.6.

## 3.3 Proposed Security Component

This section presents the proposed security component (SC). As discussed above a lightweight encryption scheme is proposed as part of the SC. The encryption scheme enforces robust identity management and access control management in Fog-CPS systems. From here onwards, the proposed scheme is referred to as the Fog-CPS security scheme. The proposed scheme has adopted the encryption and decryption Algorithms of (Odelu and Das, 2016), however some additional changes were made to the key generation Algorithms of [8] as specified below:

1. The attributes in the scheme presented here are divided into two sets i.e. secret $\mathbb{A}_S$ and shared $\mathbb{A}_K$. Therefore, the key generation process is also distributed between the certification authority (CA), FA in this case, and the Fog-CPS entities. Three algorithms namely ***Partial Key Pair Generation***, ***Final Public KeyGen*** and ***Final Secret KeyGen*** are designed for the complete key generation process. The formal construction of the key generation algorithms is different from the existing one (Odelu and Das, 2016).

2. Fog-CPS security scheme uses two elliptic curve (EC) points for each attribute instead of three as in [8]. The use of two points per attribute reduces the processing and memory overhead and make Fog-CPS scheme efficient but also secure.

3. Efficient Key Update Algorithms with limited additional overhead are introduced. The key update process only incurs the overhead of one extra element in each update. Similar to key generation process, the key update process is also split into three algorithms.

Table 3.1 Attributes Shared between CPS Devices and Fog Assist Node

| **Secret Attribute Set shared between Fog-CPS entities and FA** | | | | | |
|---|---|---|---|---|---|
| $\acute{A}_1$ | $\acute{A}_2$ | $\acute{A}_3$ | ... | ... | $\mathbb{A}_S$ |
| Device ID | IP Address | Location | Malicious Activities Reported | Number of Key Updates | Trust |
| **Shared Attribute Set shared between Fog-CPS entities** | | | | | |
| $A_1$ | $A_2$ | $A_3$ | ... | ... | $\mathbb{A}_K$ |
| Entity ID | Entity Type | Application ID | Application Type | Data Identifier | Trust |

### 3.3.1 Set of Attributes

Every Fog-CPS entity is defined by a set of attributes which are shared among the CPS devices, fog nodes and FA. Table 3.1 lists the attributes in both sets. It can be observed that the attribute set is divided into two subsets in order to maintain the privacy of sensitive attributes and to prevent the leakage of cryptographic keys. In existing ABE schemes, the user attribute set is shared with the CA. The CA generates the secret keys for the users. However, the compromise of CA can endanger the secret keys and therefore the secrecy of encrypted messages. In the proposed scheme, this challenge is addressed by dividing the attribute set $\mathbb{A}$ into two sets, namely the shared $\mathbb{A}_K$ and secret $\mathbb{A}_S$. The secret attributes $\mathbb{A}_S$ are known to the FA while the shared attributes $\mathbb{A}_K$ are known to the collaborating fog nodes and/or CPS devices.

Each CPS device or IoT device can have numerous unique attributes such as, device ID, IP address, Bluetooth device address, location (geospatial coordinates and postal addresses etc). Additionally, there are other attributes which could be shared by several devices namely, application ID, application type, data identifier, URI and URL. The attributes that could be put in two sets depend upon the Fog-CPS use case under consideration. The system analyst can carefully select the attributes for two sets after thoroughly investigating the security and trust challenges faced by such systems. As an example, a few attributes for both sets are listed in Table 3.1.

### 3.3.2 Key Pair Generation

In the proposed Fog-CPS scheme, each CPS device and fog node will have two key pairs. One is generated from secret attribute $\mathbb{A}_S$ set, the other from the shared attribute $\mathbb{A}_K$ set. As mentioned above the secret attributes are only shared with FA which registers the entities and generates the key pair $PK_{\mathbb{A}_S}/SK_{\mathbb{A}_S}$ based on $\mathbb{A}_S$. The FA subsequently publishes the public

key $PK_{\mathbb{A}_S}$ and keeps the secret key $SK_{\mathbb{A}_S}$ with itself. The registration of CPS entities with FA ensures that published public keys are authentic and do not need any further verification.

However, as the secret attributes $\mathbb{A}_S$ are only known to FA, the other collaborating CPS devices cannot verify them. To address this problem, a notion of shared/public attributes $\mathbb{A}_K$ is introduced. The shared attributes are known to collaborating Fog-CPS entities which generate the ***final public and secret keys***. The encryption and decryption would take place using ***final public and secret keys***. Such an approach is advantageous for two reasons:

1. The secret attributes are only shared with the FA and the leakage of secret keys would not risk the communication of collaborating entities.

2. The scheme is scalable because the final public keys are generated by the collaborating devices themselves without the aid of FA.

Any device can generate the final public keys of other devices. Additionally, the proposed scheme eliminates the need to have separate constructions for authentication and authorization, as only those devices which possess a shared attribute set can collaborate.

### 3.3.3 Assumptions

- CPS devices such as, smart meter and smart appliances can be compromised and leak sensitive information.

- The FA, fog nodes and cloud provider may act maliciously and try to gather information (from CPS devices, users, social media and external resources). This information can be used to generate the profile of CPS devices/users which can subsequently be used for malicious activities including targeted advertisement and spamming.

- All Fog-CPS entities will register with the FA using their secret attribute set.

- The CPS entities and fog nodes can generate the public keys, also called final public keys, of collaborating devices.

- Access policies are shared among CPS devices, FA, fog nodes and cloud.

### 3.3.4 Preliminaries

**Attribute and Access Structure:**

All CPS devices and fog nodes possess a set of attributes. Let $\mathbb{A} = \mathbb{A}_S \cup \mathbb{A}_K$ be the attribute set of each CPS device consisting of both secret $\mathbb{A}_S$ and shared $\mathbb{A}_K$ attributes. The attribute

set $\mathbb{A}_S$ can only be shared with FA whereas $\mathbb{A}_K$ can be shared with CPS devices and fog nodes. Let $\mathbb{S}_K = a_1a_2....a_n$ be the attribute string of a CPS device. The attribute string is represented with an $n$-bit string $a_1a_2....a_n$ which is defined as follows:

$$a_i = 1, \text{ if } A_i \in \mathbb{A}_K,$$
$$a_i = 0, \text{ if } A_i \notin \mathbb{A}_K,$$

For example, if $n = 6$ and $\mathbb{A}_K = \{A_1, A_2, A_4, A_6\}$, then the 6-bit string $\mathbb{S}_K$ becomes 110101.

Moreover, in the proposed scheme, AND-gate access control structure is considered for defining the access policy. The access policy $\mathbb{P}$ is defined based on the attributes in shared attribute set $\mathbb{A}_K$. Similar to device attribute string, the access policy is also represented with an $n$-bit string $\mathbb{S}_P = b_1b_2....b_n$, where

$$b_i = 1, \text{ if } A_i \in \mathbb{P},$$
$$b_i = 0, \text{ if } A_i \notin \mathbb{P},$$

For example, if $n = 6$ then policy string $\mathbb{S}_P = 101010$ meaning that the access policy $\mathbb{P}$ requires three attributes i.e. $A_1$, $A_3$ and $A_5$.

Let $\mathbb{A}_K$ be a shared attribute set and $\mathbb{P}$ be the access policy. Attribute set $\mathbb{A}_K$ satisfies the access policy, $\mathbb{P} \subseteq \mathbb{A}_K$, if and only if $a_i \in \mathbb{S}_K \geq b_i \in \mathbb{S}_P$, for all $i = 1, 2, ...., n$.

**Computational Hard Problems:**

The security of the Fog-CPS scheme is based on the computational problems described below.

**q-Generalized Diffie-Hellman (q-GDH) assumption (Boneh and Boyen, 2004)**

Given $a_1P, a_2P, ...., a_qP$ in $\mathbb{G}$ and all the subset products $(\prod_{i \in S} a_i)P \in \mathbb{G}$ for any strict subset $S \subset \{1, ..., q\}$, it is hard to compute $(a_1....a_q)P \in \mathbb{G}$, where $P$ is a base point in $E_p(a,b)$; $a_1, a_2, ..., a_q \in Z_p^*$. Since the number of subset products (elliptic curve scalar point multiplications) is exponential in $q$, access to all these subset products is provided through an oracle. For a vector $a = (a_1, ..., a_q) \in (Z_p)^q$, define $\mathscr{O}_{p,a}$ to be an oracle that for any strict subset $S \subset \{1, ...., q\}$ responds with $\mathscr{O}_{p,a}(S) = (\prod_{i \in S} a_i) \in \mathbb{G}$.

**Definition 1** (q-GDH assumption). *The $(t, q, \varepsilon) - GDH$ assumption is satisfied in $\mathbb{G}$, if for all $t$-time algorithms $A$, the advantage $Adv_{A,q}^{GDH} = Pr[\mathscr{A}^{\mathscr{O}_{P,a}} = (a_1...a_q)P] < \varepsilon$, where $a = (a_1, ...., a_q) \in (Z_p)^q$ and for any sufficiently small $\varepsilon > 0$ .*

**q-Diffie-Hellman Inversion (q-DHI) problem (Boneh and Boyen, 2004)**
Given a $(q+1)$-tuple $(P, xP, x^2P, ..., x^qP) \in \mathbb{G}^{q+1}$, the problem is to compute $(1/x)P \in \mathbb{G}$ where $x \in Z_p^*$.

**Definition 2** (q-DHI assumption). $\mathbb{G}$ *satisfies the* $(t, q, \varepsilon)$*-DHI assumption, if for all* $t$*-time algorithms* $\mathscr{A}$*, the advantage becomes* $Adv_{A,q}^{GDH} = Pr[\mathscr{A}(P, xP, x^2P, ...x^qP) = (1/x)P] < \varepsilon$ *for any sufficiently small* $\varepsilon > 0$*, where the probability is over the random choice of* $x$ *in* $Z_p^*$ *and the random bits of* $\mathscr{A}$*.*

### 3.3.5 Fog-CPS Scheme Application in Fog-SGC Scenario

This section discusses the application of the proposed scheme for a Fog-SGC system. A use case where a smart home device reports meter readings data $D$ to the NAN device. Figure 3.3 illustrates the communication between smart home device, NAN device and FA. Initially, all entities namely CPS devices and fog nodes will register with FA based on their secret attribute sets $\mathbb{A}_S$. FA subsequently publishes their partial public keys $PK_{\mathbb{A}_S}$.

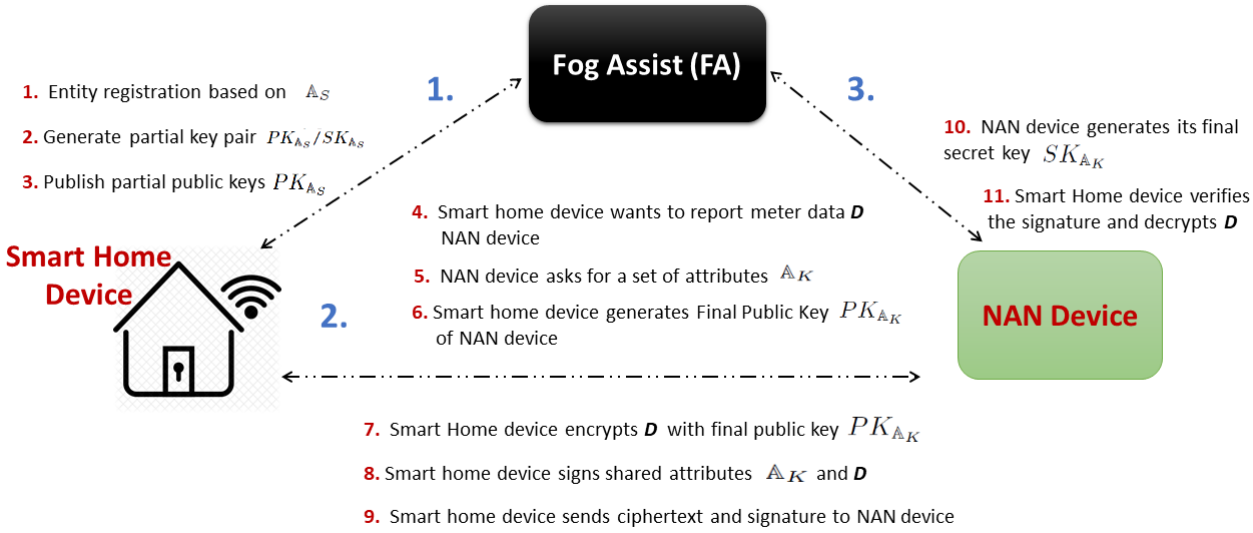


Fig. 3.3 Fog-CPS Scheme Application in Fog-SGC Scenario

Next, the smart home device sends meter data store request to NAN device. Upon receiving the request, the NAN device asks for a set of attributes $\mathbb{A}_K$. Subsequently, the smart home device generates the final public key $PK_{\mathbb{A}_K}$ of NAN device and encrypts $D$ using that key. After that smart home device signs the shared attributes $\mathbb{A}_K$ using its secret key. The smart home device sends $CT$ and signature $\sigma$ to the NAN device.

Following this, the NAN device generates the final secret key $SK_{\mathbb{A}_K}$ based on shared attributes set $\mathbb{A}_K$. If the NAN device successfully verifies the $\sigma$ and decrypts the $CT$, it gets an assurance that the smart home possesses the required attributes and stores $D$. The interaction between all other entities will be similar as between the smart home device and the NAN device.

### 3.3.6 Fog-CPS Description

The proposed Fog-CPS scheme consists of eight algorithms which are discussed below. It is assumed that the elliptic curve group parameters are pre-shared with all entities in the system and they will use them to generate the keys.

**1. Partial Key Pair Generation** $(\lambda, \mathbb{A}_S) \rightarrow PK_{\mathbb{A}_S}/SK_{\mathbb{A}_S}$

Every entity in the Fog-CPS i.e. CPS devices and fog nodes will register with the FA based on a secret set of attributes. The FA will execute the *KeyGen* algorithm to generate their key pair. This algorithm takes as input the security parameter $\lambda$ and a set of secret attributes $\mathbb{A}_S$. It outputs the partial public/secret key pair $PK_{\mathbb{A}_S}/SK_{\mathbb{A}_S}$. The FA will publish the partial public key $PK_{\mathbb{A}_S}$ and sends the partial secret key $SK_{\mathbb{A}_S}$ to the CPS device. For initial communication, Fog-CPS entities can use the partial public keys. The partial public keys guarantee that Fog-CPS entities are legitimate and registered with FA.

**2. Final Public KeyGen** $(PK_{\mathbb{A}_S}, \mathbb{A}_K) \rightarrow PK_{\mathbb{A}_K}$

The second pair of keys namely, final public and secret keys are generated from the shared attribute set. The final public keys can be generated by any CPS device or fog node that shares a set of attributes with some other entity. This algorithm generates the final public key of a CPS device. It takes as input the public key $PK_{\mathbb{A}_S}$ generated over the secret attribute set $\mathbb{A}_S$ and the shared attribute set $\mathbb{A}_K$.

**3. Final Secret KeyGen** $(SK_{\mathbb{A}_S}, \mathbb{A}_K) \rightarrow SK_{\mathbb{A}_K}$

This algorithm generates the final secret key of a CPS device. It takes as input the secret key generated over the attribute set $\mathbb{A}_S$, and the shared attribute set $\mathbb{A}_K$.

**4. Encrypt** $(PK_{\mathbb{A}_K}, \mathbb{P}, M) \rightarrow CT$

The encryption algorithm takes as input the final public key $PK_{\mathbb{A}_K}$, access policy $\mathbb{P}$ and a message $M$. It outputs a ciphertext $CT$.

**5. Decrypt** $(SK_{\mathbb{A}_K}, \mathbb{P}, CT) \rightarrow M$

The decryption algorithm takes as input the final secret key $SK_{\mathbb{A}_K}$, access policy $\mathbb{P}$ and , and outputs the plaintext message $M$ if $SK_{\mathbb{A}_K}$ statisfies $\mathbb{P}$.

**6. Partial Key Pair Update**$(\lambda, \acute{\mathbb{A}_S}) \rightarrow \acute{PK}_{\mathbb{A}_S}/\acute{SK}_{\mathbb{A}_S}$

This algorithm takes the security input and an updated set of secret attributes $\mathbb{A}_S$ as input and generates a new $\acute{PK}_{\mathbb{A}_K}/\acute{SK}_{\mathbb{A}_K}$ partial key pair for the CPS device.

**7. Final Keys Update**$(\acute{PK}_{\mathbb{A}_S}, \acute{\mathbb{A}_K}) \rightarrow \acute{PK}_{\mathbb{A}_K}/\acute{SK}_{\mathbb{A}_K}$

If the shared attributes of a CPS device are updated, then the final public and secret keys need to be regenerated. These algorithms take the updated set of shared attributes $\acute{\mathbb{A}_S}$ as input and generates the new final public and secret keys $\acute{PK}_{\mathbb{A}_K}/\acute{SK}_{\mathbb{A}_K}$ of the CPS device.

**8. KeyRevoke**

Similar to existing ABE schemes, the keys are revoked by the FA. However, the keys can also be revoked due to the malicious behaviour of CPS devices. For instance, a fog node can request a key revocation based on the malicious activities of compromised CPS devices. Three cases for key revocation are identified, 1) legitimate revoke, and 2) malicious activity and 3) attributes update. The details of the three key revocation cases are further discussed in the following section.

**Correctness.** The Fog-CPS Scheme must satisfy the following property. For any $PK_{\mathbb{A}_K}/SK_{\mathbb{A}_K}$ key pair if $\mathbb{P} \subseteq \mathbb{A}_K$, the decryption algorithm always outputs the original plaintext $M$. Otherwise, the cipher-text $E_{PK_{\mathbb{A}_K}}[\mathbb{P}, M]$ cannot be decrypted using the secret key $SK_{\mathbb{A}_K}$.

## 3.4 Mathematical Construction of Fog-CPS Scheme

In this section, the construction of all algorithms is presented.

### 3.4.1 Partial Key Pair Generation$(\lambda, \mathbb{A}_S) \rightarrow PK_{\mathbb{A}_S}/SK_{\mathbb{A}_S}$

Initially, all entities namely smart meters and fog nodes register with FA based on their secret attribute sets $\mathbb{A}_S$. The FA executes Algorithm 1 to generate the partial key pair. *KeyGen* algorithm takes as input the security parameter $\lambda$ and a set of secret attributes $\mathbb{A}_S$. The security parameter $\lambda$ is generally used in all encryption algorithms and it consists of a long string of 1s. The security parameter defines the length of the secret keys and message in an encryption scheme. In this case, the $\lambda$ is dependent on the chosen finite field which is 512 bits in case of supersingular curve SS512. Algorithm outputs the partial public/secret key

**Algorithm 1** Partial Key Pair Generation

1: **Input:** Security parameter $\lambda$, secret attribute set $\mathbb{A}_S$.
2: **Output:** $PK_{\mathbb{A}_S}/SK_{\mathbb{A}_S}$
$PK_{\mathbb{A}_S} = \{\mathbb{G}, P_i, H_1\}, \forall i = 0, 1, ..., t.$
$SK_{\mathbb{A}_S} = \{s\}$
3: Choose elliptic curve group $\mathbb{G}$, where $P$ is a base point on the elliptic curve $E_p(a,b)$ defined over the finite field $Z_p$.
4: Choose a one-way collision resistance hash function, $H_1$ defined as:

$$H_1 : \{0,1\}^* \rightarrow Z_p^*,$$

5: Create a secret random number $r \in Z_p$.
6: Map $t < n$ attributes in secret set $\mathbb{A}_S$ to $Z_p$ using hash function $H_1$ and compute secret number $s_1$.

$$s_i = H_1(i) \pmod{\text{p}}, \forall i \in \mathbb{A}_S, \tag{3.1}$$

$$\acute{s} = \sum_{i=1}^{t} s_i, \tag{3.2}$$

$$s = r\acute{s} \tag{3.3}$$

7: Next compute public key components $P_i$ as:

$$P_i = s^i P, \quad \forall i = 0, 1, ....., t \tag{3.4}$$

pair $PK_{\mathbb{A}_S}/SK_{\mathbb{A}_S}$. Subsequently, the FA publishes the partial public key $PK_{\mathbb{A}_S}$ and sends the partial secret key $SK_{\mathbb{A}_S}$ to the registering device.

### 3.4.2 Final Public KeyGen$(PK_{\mathbb{A}_S}, \mathbb{A}_K) \rightarrow PK_{\mathbb{A}_K}$

After registration with FA, the smart home device sends a data store request to the NAN device. Upon receiving the request, the NAN device asks for a set of attributes $\mathbb{A}_K$. Subsequently, the smart home device generates the final public key of the NAN device by executing Algorithm 2. This algorithm generates the final public key of a CPS device. It takes as input the partial public key $PK_{\mathbb{A}_S}$ generated over the secret attribute set $\mathbb{A}_S$ and the shared attribute set $\mathbb{A}_K$. It outputs the final public key $PK_{\mathbb{A}_K}$.

**Algorithm 2** Final Public Key Generation

1: **Input:** Partial Public Key $PK_{\mathbb{A}_S}$ and shared attribute set $\mathbb{A}_K$.
2: **Output:** $PK_{\mathbb{A}_K} = \{U_i\}, \forall i = 0,1,...,t$.
3: Let $\mathbb{S}_K = a_1 a_2 .... a_t$ be the device attribute string over shared attribute set.
4: Map $t$, $t < n$ attributes in shared set $\mathbb{A}_K$ to $Z_p$ using hash function $H_1$ and compute $k$.

$$\begin{aligned} k_i &= H_1(i) \pmod{\text{p}}, \forall i \in \mathbb{A}_K, \\ k &= \sum_{i=1}^{t} k_i, \end{aligned} \tag{3.5}$$

5: Next, compute final public key components $U_i$, $\forall i = 0,1,...,t$ as:

$$U_i = k^i P_i, \tag{3.6}$$

### 3.4.3 Final Secret KeyGen$(SK_{\mathbb{A}_S}, \mathbb{A}_K) \rightarrow SK_{\mathbb{A}_K}$

The NAN device executes Algorithm 3 to generate the final secret key $SK_{\mathbb{A}_K}$ corresponding to a shared attribute set $\mathbb{A}_K$. It takes as input the secret key generated over the secret attribute set $\mathbb{A}_S$ and the shared attribute set $\mathbb{A}_K$. It outputs the final secret key $SK_{\mathbb{SK}_K}$.

### 3.4.4 Encrypt$(PK_{\mathbb{A}_K}, \mathbb{P}, M) \rightarrow CT$

Next, the smart home device encrypts $M$ using the final public key $PK_{\mathbb{A}_K}$ of the NAN device and signs the shared attributes $\mathbb{A}_K$ using its secret key. The smart home device sends $CT$ and signature $\sigma$ to the NAN device. The encryption algorithm takes as input the final public key $PK_{\mathbb{A}_K}$, access policy $\mathbb{P}$ and a message $M$. It outputs a ciphertext $CT$. Algorithm 4 presents the encryption procedure in detail.

**Proposition 1.** *From Equations (3.8) and (3.13, a new polynomial can be calculated as:*

$$F(x, \mathbb{S}_K, \mathbb{S}_P) = \frac{f(x, \mathbb{S}_P)}{f(x, \mathbb{S}_K)} = \prod_{i=1}^{t} (x + H_4(i))^{a_i - b_i} \tag{3.18}$$

*It can easily be verified that* $\frac{f(x,\mathbb{S}_P)}{f(x,\mathbb{S}_K)}$ *is a polynomial function in x, if and only if* $\mathbb{P} \subseteq \mathbb{A}_K$. The encryption algorithm and the secret key generation algorithm are designed in such a way that $\frac{f(x,\mathbb{S}_P)}{f(x,\mathbb{S}_K)}$ must be a polynomial for a successful decryption.

**Algorithm 3** Final Secret Key Generation

1: **Input:** Partial secret key $SK_{\mathbb{A}_S}$ and shared attribute set $\mathbb{A}_K$.
2: **Output:** Final secret key $SK_{\mathbb{A}_K} = \{u_1\}$.
3: Compute secret number $\alpha$ as follows:

$$\alpha = sk, \tag{3.7}$$

where $s$ is the secret key component from $SK_{\mathbb{A}_S}$ and $k$ is computed similar to Equation 3.5 in Algorithm 2.
4: Next, compute Eq. 3.8, $f(\alpha, \mathbb{S}_K)$ which is an $t$-degree at most polynomial in $Z_p[x]$.

$$f(\alpha, \mathbb{S}_K) = \prod_{i=1}^{t} (\alpha + H_1(i))^{1-a_i} \tag{3.8}$$

where $i$ is an attribute in the $\mathbb{A}_K$.
5: Pick two random numbers $r_u, t_u \in Z_p$. Compute $s_u$ such that the following condition holds.

$$\frac{1}{f(\alpha, \mathbb{S}_K)} = s_u - r_u t_u \pmod{\text{p}}$$

$$s_u = \frac{1}{f(\alpha, \mathbb{S}_K)} + r_u t_u \tag{3.9}$$

6: Next, compute secret key component $u_1$.

$$u_1 = s_u - r_u t_u \pmod{\text{p}}, \tag{3.10}$$

### 3.4.5 Decrypt$(SK_{\mathbb{A}_K}, \mathbb{P}, CT) \rightarrow M$

Following this, the NAN device verifies the $\sigma$ and decrypts the $CT$. Upon successful decryption, it gets an assurance that the smart home device possesses the required attributes and stores $D$. Algorithm 5 elaborates the decryption process. The decryption algorithm takes as input the secret key $SK_{\mathbb{A}_K}$ and ciphertext $CT$ and outputs the plaintext message $M$.

First of all, the decryption key $r_m P$ is computed. Subsequently, $\acute{c} = H_2(KDF(r_m P)) \oplus C_r$, $\acute{M} = C_m \oplus H_3(\acute{c})$ and $\acute{r_m} = H_1((\mathbb{P}, \acute{M}), \acute{c})$ are computed. Next, the equality $r_m P = \acute{r_m} P$ is checked. If equality holds, $\acute{M}$ is treated as the original plaintext $M$, otherwise outputs null $(\perp)$.

**Algorithm 4** Encryption

1: **Input:** $PK_{\mathbb{A}_K}$, $M$, **and** $\mathbb{P}$, $\mathbb{P} \subseteq \mathbb{A}_K$ .
2: **Output:** $CT = \{\mathbb{P}, U_{m,i}, C_1, C_r, C_m\}$.
3: Create a random number $c \in \{0,1\}^{l_r}$ and compute

$$r_m = H_1(\mathbb{P}, M, c), \tag{3.11}$$
$$k_m = KDF(r_m P), \tag{3.12}$$

where $KDF$ is a key derivation function which takes the new elliptic curve point $r_m P$ and generates a secret key $k_m$.

4: Let $\mathbb{S}_P = b_1 b_2 .... b_t$ be the access policy string. Compute the corresponding $(t-1)$ degree at most polynomial function $f(x, \mathbb{S}_P)$ in $Z_p[x]$ as

$$f(x, \mathbb{S}_P) = \prod_{i=1}^{t} (x + H_1(i))^{1-b_i} \tag{3.13}$$

Let $c_i$ denotes the coefficient of $x^i$ in the polynomial $f(x, \mathbb{S}_P)$.

5: Choose two one-way collision resistance hash functions, $H_2$ and $H_3$ defined as:

$$H_2 : \{0,1\}^* \rightarrow \{0,1\}^{l_r},$$
$$H_3 : \{0,1\}^* \rightarrow \{0,1\}^{l_m},$$

where $l_r$ is the length of a random string, $l_m$ is the length of message $M$, $\{0,1\}^*$ is a binary string of arbitrary length, and $\{0,1\}^l$ is a binary string of length $l$. The length of the hash value is the same as the length of a random string $r$ and similarly the hash value will be the same size as the message $M$.

6: Next, the $CT$ which consists of three components $C_1$, $C_r$ and $C_m$ is computed. $C_1$ is a point on elliptic curve which is computed from the polynomial $f(x, \mathbb{S}_P)$ and $U_i$ components in $PK_{\mathbb{A}_K}$ corresponding to attributes in $\mathbb{P}$. $C_1$ is computed as follows:

$$U_{m,i} = r_m U_i, \qquad i = 1, 2, ..., t - |\mathbb{P}|, \tag{3.14}$$

$$C_1 = r_m \sum_{i=0}^{t} c_i U_i = r_m f(\alpha, \mathbb{P}) P, \tag{3.15}$$

Next, $C_r$ is computed by a XOR operation on $k_m$ and $c$ computed in Step 3.

$$C_r = H_2(k_m) \oplus c, \tag{3.16}$$

Lastly, $C_m$ is computed by a XOR operation on $c$ and message $M$.

$$C_m = H_3(c) \oplus M, \tag{3.17}$$

**Algorithm 5** Decryption

1: **Input:** $SK_{\mathbb{A}_K}$, $CT = \{\mathbb{P}, U_{m,i}, C_1, C_r, C_m\}$.
2: **Output:** $M$ **or** $\perp$
3: Compute $V$ as follows:

$$\begin{aligned} V &= u_1 C_1, \\ &= (s_u - r_u t_u) r_m \sum_{i=0}^{t} c_i U_i \\ &= \left(\frac{1}{f(\alpha, \mathbb{A}_K)} + r_u t_u - r_u t_u\right) r_m f(\alpha, \mathbb{P}) P \\ &= r_m \frac{1}{f(\alpha, \mathbb{A}_K)} f(\alpha, \mathbb{P}) P \\ &= r_m F(\alpha) P. \end{aligned} \tag{3.19}$$

4: Compute $c_i = a_i - b_i$ for $i = 1, 2, ...., t$. Let $F(x, \mathbb{S}_K, \mathbb{S}_P)$ be the $(t - |\mathbb{P}|)$ degree at most polynomial function in $Z_p[x]$ defined as

$$F(x) = F(x, \mathbb{S}_K, \mathbb{S}_P) = \prod_{i=1}^{t-|\mathbb{P}|} \big(x + H_1(i)\big)^{c_i} \tag{3.20}$$

and $\acute{c}_i$ be the coefficient of $x^i$ in the polynomial $F(x)$. It is clear that $\acute{c}_0 \neq 0$ .
5: Next, compute

$$\begin{aligned} W &= \sum_{i=1}^{(t-|\mathbb{P}|)} \acute{c}_i U_{m,i} \\ &= r_m \Big( \sum_{i=1}^{(t-|\mathbb{P}|)} \acute{c}_i \alpha^i \Big) P \\ &= r_m \Big( \sum_{i=1}^{(t-|\mathbb{P}|)} \acute{c}_i \alpha^i + \acute{c}_0 - \acute{c}_0 \Big) P \\ &= r_m (F(\alpha) - \acute{c}_0) P \\ &= r_m F(\alpha) P - r_m \acute{c}_0 P \end{aligned} \tag{3.21}$$

6: Compute the decryption key $r_m P$ as $\frac{1}{\acute{c}_0}(V-W)$.

$$\begin{aligned} r_m P &= \frac{1}{\acute{c}_0}(V-W) \\ &= \frac{1}{\acute{c}_0}(r_m F(\alpha)P - (r_m F(\alpha)P - r_m \acute{c}_0 P)) \\ &= \frac{r_m F(\alpha) - r_m F(\alpha) + r_m \acute{c}_0}{\acute{c}_0} P \\ &= r_m P \end{aligned}$$

7: Next, compute,

$$\begin{aligned} \acute{c} &= H_2(KDF(r_m P)) \oplus C_r, \\ \acute{M} &= C_m \oplus H_3(\acute{c}), \\ \acute{r}_m &= H_1(\mathbb{P}, \acute{M}, \acute{c}) \end{aligned}$$

8: Finally, verify $r_m P = \acute{r}_m P$. If the condition holds, treat $\acute{M}$ as original plaintext message $M$. Otherwise, output null ($\perp$).

**Remark.** *If $\mathbb{A}_K = \mathbb{P}$, then $F(x) = 1$, which is a constant polynomial. This implies that $\frac{f(\alpha,\mathbb{P})}{f(\alpha,\mathbb{A}_K)} = F(\alpha) = 1$. Hence, $V = r_m P$, and in this case, Equation 3.21 is not computed.*

### 3.4.6 Partial Key Pair Update$(\lambda, \acute{\mathbb{A}_S}) \rightarrow P\acute{K}_{\mathbb{A}_S}/S\acute{K}_{\mathbb{A}_S}$

If the secret attributes of a CPS device are changed, then all the keys need to be updated. The key update procedure will start by regenerating the partial public/secret $PK_{\mathbb{A}_S}/SK_{\mathbb{A}_S}$ key pair. In partial key pair update procedure, Algorithm 6 is executed. It takes the updated set of secret attributes $\mathbb{A}_S$ as input and generates a new $PK_{\mathbb{A}_S}/SK_{\mathbb{A}_S}$ partial key pair for the CPS device. Subsequently, the final public and secret keys are also regenerated.

### 3.4.7 Final Keys Update$(P\acute{K}_{\mathbb{A}_S}, \acute{\mathbb{A}_K}) \rightarrow P\acute{K}_{\mathbb{A}_K}/S\acute{K}_{\mathbb{A}_K}$

If the shared attributes of a CPS device are updated, then the final public and secret keys need to be regenerated. In this case, Algorithms 7 and 8 are executed to generate the new final public and secret keys of the Fog-CPS entities. These algorithms take the updated set of shared attributes $\acute{\mathbb{A}_k}$ as input and generates the new final public and secret keys $P\acute{K}_{\mathbb{A}_K}/S\acute{K}_{\mathbb{A}_K}$ of the CPS device.

**Algorithm 6** Partial Key Pair Update

1: **Input:** Secret $\mathbb{A}_S$ set.
2: **Output:** New partial key pair $PK_{\mathbb{A}_S}/SK_{\mathbb{A}_S}$.
3: Increment the counter $c$ for revoked keys.
4: If the attributes are the same, the previous computations over $t$ attributes are considered.
5: Else, perform calculations for attributes from $i = t$ to $t \pm 1$.
6: Map $t \pm 1$ attribute in $\mathbb{A}_S$ to $Z_p$ using Eq. (1) and compute secret number $\acute{s}$.

$$\acute{s} = \acute{s} + s_{t\pm1}, \tag{3.22}$$

7: Next compute $s$ using Eq. (3).
8: Next compute public key component $P_i$ for $i = t \pm 1$ attribute using Eq. (4)

**Algorithm 7** Final Public Key Update

1: **Input:** Shared $\mathbb{A}_K$ attribute set.
2: **Output:** New final public key $PK_{\mathbb{A}_K}$.
3: If the attributes are the same, the previous computations over $t$ attributes are considered.
4: Else, perform calculations for attributes from $i = t$ to $t \pm 1$.
5: Map $(t \pm 1)$ attribute in shared set $\mathbb{A}_K$ to $Z_p$. Apply hash function $H_1$ and compute $k_i$ using Eq. (5). Then compute k:

$$k = k + k_{t\pm1} \tag{3.23}$$

6: Next, compute final public key component $U_i$, for $(t \pm 1)$ attribute using Eq. (7)

**KeyRevoke**

Similar to existing ABE schemes, the keys are revoked by the CA, but in our application scenario FA. However, the keys can also be revoked due to the malicious behaviour of CPS devices. Three cases for key revocation have been identified :

1. **Legitimate Revoke:** In the first case, both key pairs can be revoked due to system update, expiration date and scheduled maintenance of Fog-CPS system.

2. **Malicious Activity:** In the second case, the key revocation may take place due to the malicious behaviour which might be observed and/or reported by FA, fog nodes and CPS devices. Two cases of malicious activity are considered below:
   **- Malicious FA:** The compromise of partial key pair would not trigger key revocation.
   **- Malicious CPS Devices:** The compromise of a CPS device would trigger revocation of partial key pair.

**Algorithm 8** Final Secret Key Update

1: **Input:** Shared $\mathbb{A}_K$ attribute set.
2: **Output:** New final secret key $SK_{\mathbb{A}_K}$
3: If the attributes are the same, the previous computations over $t$ attributes are considered.
4: Else, perform calculations for attributes from $i = t$ to $t \pm 1$.
5: Compute secret number $\alpha$ using Eq. (8)
6: Next, compute $f(\alpha, \mathbb{A}_K)$:

$$f(\alpha, \mathbb{A}_K) = f(\alpha, \mathbb{A}_K) \cdot (\alpha + H_1(t \pm 1))^{1-a_{t\pm1}}, \tag{3.24}$$

where polynomial $f(\alpha, \mathbb{A}_K)$ has been computed over $t$ attributes in Eq. 3.8.
7: Pick two random numbers $r_u, t_u \in Z_P$ and then Compute $s_u$ using Eq. (10)
8: Next, compute secret key component $u_1$ using Eq. (11)

3. **Attributes Update:** In the third case, the change in the attribute set can trigger a key revocation. For example, if the secret key is compromised or leaked, then again keys should be revoked and regenerated.

In key revocation, FA revokes the existing partial key pair and generates new keys in first two cases i.e. legitimate revoke and malicious activity. In the third case, the keys are regenerated as discussed in section 3.4.6 and 3.4.7. As the generation of final public and secret key is dependent upon the partial key pair generated over secret attributes. So, the revocation of partial key pair requires the revocation of final public and secret keys. As a result, Algorithms 6, 7, 8 are designed. The proposed key update algorithms are lightweight as each revocation only incurs the overhead of one extra key component. In each subsequent key update, the $t$ attributes counter is incremented by one.

## 3.5 Theoretical Security Analysis and Evaluation

As mentioned in the previous sections, in the proposed Fog-CPS security scheme, each entity possesses two key pairs namely, partial and secret. So keeping that in view, the security of the proposed scheme is carefully analyzed to ensure security against following attacks:

- computing the final secret key $SK_{\mathbb{A}_K}$ from partial secret key $SK_{\mathbb{A}_S}$
- computing the partial/final secret keys from partial/final public keys
- computing the final secret key $SK_{\mathbb{A}_K}$ from multiple ciphertexts (i.e chosen ciphertext attack).

The Fog-CPS scheme is secure against the above mentioned attacks due to the q-Diffie–Hellman Inversion (q-DHI) problem Boneh and Boyen (2004), Elliptic Curve Discrete Logarithm Problem (ECDLP), and the robustness of the hash functions.

Additionally, the robustness of the proposed scheme is evaluated by a selective security game which is based on two fundamental security notions of encryption schemes namely, indistinguishability of messages and the collision resistance against secret keys. Message indistinguishability is an important security property of many encryption schemes. Given the ciphertext and the encryption key, the adversary cannot tell apart two same length but different messages encrypted under the scheme, even if he chose the messages himself. With collision resistance, the attackers cannot pool their secret key components corresponding to a set of attributes to generate a new key which otherwise cannot be generated from their own attributes. The selective security game also models the interactions between the adversary $\mathscr{A}$ and the challenger $\mathscr{B}$. But before presenting the security analysis against above mentioned attacks, the notion of collision resistance as presumed in this scheme is discussed.

**- Collision Resistance against Secret Keys**

The proposed scheme does not follow the same conventional attribute sharing as the existing ABE schemes. Attributes are only shared between two CPS devices and the FA node. So, the collision attack as presumed in existing schemes does not apply in this case. In other words, the pooling of attributes and secret key components (i.e. the collision attack) from several adversaries who do not share the attributes would not benefit in generating the secret keys. Precisely, for the security of the proposed scheme, the definition of collision resistance is modified. In this case, it is essential to prevent a device from generating the final secret key $SK_{\mathbb{A}_K}$ of another device.

### 3.5.1 Chosen Ciphertext Attack with in a Selective Security Game Model

In the selective security game, the collision-resistance is handled by allowing the adversary $\mathscr{A}$ to request multiple secret keys, once it declares the challenge. To capture the collision-resistance, multiple secret key queries can be issued by an adversary $\mathscr{A}$ after the challenge phase. Following this, the game between the adversary $\mathscr{A}$ and a challenger $\mathscr{B}$ is described.

- **Initialization:** Firstly, $\mathscr{A}$ declares a challenge as an $n$-bit access policy string $\acute{\mathbb{P}}$ corresponding to $\acute{\mathbb{A}_K}$ and sends it to the challenger $\mathscr{B}$.
- **Final Key Pair Generation:** Secondly, $\mathscr{B}$ runs Algorithms 1, 2 and 3 with the attribute sets $\mathbb{A}_S$, $\mathbb{A}_K$ to generate the partial key pair $PK_{\mathbb{A}_S}/SK_{\mathbb{A}_S}$ and final public $PK_{\mathbb{A}_K}$ and secret $SK_{\mathbb{A}_K}$ keys. It then gives $PK_{\mathbb{A}_K}/SK_{\mathbb{A}_K}$ to $\mathscr{A}$.

- **Query:** Thirdly, $\mathscr{A}$ requests for multiple key pairs and decryptions as described below:
  – $\mathscr{A}$ queries for the key pair $(PK_{\acute{\mathbb{A}_K}}, SK_{\acute{\mathbb{A}_K}})$ for an attribute set $\acute{\mathbb{A}_K}$ which does not fulfil the access policy $\acute{\mathbb{P}}$. $\mathscr{B}$ subsequently generates the key pair $(PK_{\acute{\mathbb{A}_K}}, SK_{\acute{\mathbb{A}_K}})$ and sends it to $\mathscr{A}$.
  - $\mathscr{A}$ also requests the decryption corresponding to the ciphertext $CT = E[\acute{\mathbb{P}}, \acute{M}]$.
- **Challenge:** Fourthly, the adversary $\mathscr{A}$ outputs two messages $(M_0, M_1)$ for challenge. As mentioned in the Query phase that $\mathscr{A}$ must not have queried for a secret key corresponding to an attribute set $\mathbb{A}$ which satisfies $\acute{\mathbb{P}} \subseteq \mathbb{A}_K$. The challenger $\mathscr{B}$ responds by picking a random bit $\acute{b} \in \{0,1\}$ and compute the challenge ciphertext $CT = E[\acute{\mathbb{P}}, M_{\acute{b}}]$.
- **Guess:** Lastly, the adversary $\mathscr{A}$ outputs a guess $\acute{b}_g$ of $\acute{b}$. $\mathscr{A}$ wins the game if $\acute{b}_g = \acute{b}$.

In this game, the advantage $\varepsilon$ of the adversary $\mathscr{A}$ is defined by:

$$\varepsilon = Pr[\acute{b}_g, \acute{b}] - \frac{1}{2}$$

**Remark.** *As $\varepsilon$ is a negligible function of $\lambda$ in the above security game, the proposed scheme is said to be $(t, q_s, q_d, \varepsilon)$-selectively secure against a chosen-ciphertext attack, if for all t-polynomial time adversaries who make the $q_s$ key pair and $q_d$ decryption queries at most.*

### 3.5.2 Key Generation Analysis

In this section, the difficulty of driving the partial/final secret keys from their respective public keys, and final secret key derivation from multiple ciphertexts and partial secret key is analyzed. Additionally, the computational difficulty of guessing the attributes and subsequently generating the secret keys is also discussed.

#### Partial/Final Secret Key Guessing

The partial and final secret keys in Algorithms 1 and 3 are generated based on the secret and shared attributes which are mapped to $Z_p$. The success probability of guessing an attribute is equivalent to the complexity of hashing algorithm $H_1$ i.e., $2^{n/2}$ (birthday paradox). For the partial secret key $SK_{\mathbb{A}_S}$, the adversary should guess all attributes in set $\mathbb{A}_S$ and the secret random number $r$. The secret numbers $s_i$ which are used in partial key pair generation cannot be derived by collision attack due to its complexity. Precisely, the computational complexity is of the order of number of attributes for hash function and random guessing. This also applies to final public and secret key generation algorithms whereby the shared attributes are hashed and subsequently used in key generation. Additionally, the assumption that each

entity possesses a unique set of secret and shared attributes with no overlapping with the attribute set of other entities makes attribute guessing more difficult.

### Partial/Final Secret Key Generation

The partial secret key $SK_{\mathbb{A}_S}$ of a CPS device cannot be guessed due to the difficulty of deriving the secret key components $s_i \in \mathbb{A}_S$ and $r \in Z_p$ in Algorithm 1. So, in order to generate/guess the final secret key, the adversary needs to know the secret key $SK_{\mathbb{A}_S}$, shared attribute set $\mathbb{A}_K$ and three secret numbers $\alpha, r_u, t_u$. $\alpha$ is computed from secret components $s$ and $k$ in algorithm 3.4.2 whereas $r_u, t_u$ are random numbers. The secret component $s$ can only be computed and/or known if both FA node and the CPS device are compromised. The compromised device can leak the shared attribute set and the final secret key $SK_{\mathbb{A}_K}$.

**Theorem 1.** *The proposed scheme is secure against an adversary $\mathscr{A}$ with knowledge of the shared attribute set $\mathbb{A}_K$ for deriving the final $SK_{\mathbb{A}_K}$ secret key by collision attack.*

*Proof.* Having the knowledge of $\mathbb{A}_K$ is not enough for generating the $SK_{\mathbb{A}_K} = u_1$, where

$$u_1 = s_u - r_u t_u \text{ (mod p)}, \tag{3.25}$$

From Eq. 3.9 in Algorithm 3,

$$\frac{1}{f(\alpha, \mathbb{A}_K)} = s_u - r_u t_u \text{ (mod p)}. \tag{3.26}$$

where $r_u$ and $t_u$ are random numbers. The condition in Eq. 3.26 only holds if $s_u$ and $\alpha$ are known and subsequently the values of $r_u t_u$ can be computed. All these values can then be used to solve Eq. 3.25. Another solution to Eq. 3.25 is to correctly guess the random numbers $r_u$, $t_u$ and compute $\alpha$. The difficulty of computing $\alpha$ is already explained in preceding paragraphs. Hence, generating $SK_{\mathbb{A}_K}$ without knowing secret components $(r_u, t_u)$, $s_u$ and $\alpha$ is computationally infeasible for an adversary. □

### Computing the Secret Keys from Public Keys

It is underlined that the secret keys either partial/final cannot be computed from their respective public keys due to the intractability of the Elliptic Curve Discrete Logarithm Problem (ECDLP). Given two points $P, Q \in E(F_q)$, the ECDLP problem is to find an integer $x$, if it exists, such that $Q = xP$. Following the same notion, the problem is to compute partial/final secret $SK_{\mathbb{A}_S}/SK_{\mathbb{A}_K}$ keys from public keys $PK_{\mathbb{A}_S}/PK_{\mathbb{A}_K}$. Like in case of $SK_{\mathbb{A}_S}$, given the

$PK_{\mathbb{A}_S} = \{P_i = s^i P\}$ for all attributes in $\mathbb{A}_S$, the problem is to compute $s^i$ from its corresponding public key $P_i$ component. The ECDLP problem has to be solved for all attributes in a given attribute set. The same applies to the final secret key $SK_{\mathbb{A}_K}$ generation from $PK_{\mathbb{A}_K}$. For the $SK_{\mathbb{A}_K}$, ECDLP is to compute $k_i$ from given $U_i$ and $P_i$. Precisely, due to the intractability of ECDLP, it is not feasible to compute secret keys from public keys.

### Computing the Decryption Key from Ciphertext

Additionally, the proposed scheme is secure against an adversary for deriving the decryption key $r_m P$ from the ciphertext $CT = \{\mathbb{P}, U_{m,i}, K_{1m}, C_1, C_{\sigma m}, C_m\}$.

**Theorem 2.** *Given the ciphertext $CT = \{\mathbb{P}, U_{m,i}, K_{1m}, C_1, C_r, C_m\}$, it is hard to compute decryption key $r_m P$.*

*Proof.* A ciphertext $CT$ corresponding to the access policy $\mathbb{P}$ consists of the following parameters:

$$\begin{aligned} U_{m,i} &= r_m P_i, \qquad i = 1,2,...,n-|\mathbb{P}|, \\ C_1 &= r_m \sum_{i=0}^{n} f_i U_i = r_m f(\alpha, \mathbb{P}) P, \\ C_r &= H_2(k_m) \oplus c, \\ C_m &= H_3(c) \oplus M \end{aligned}$$

Since $\sum_{i=1}^{n-|\mathbb{P}|} U_{m,i} = rm(f(\alpha,\mathbb{P}) - f_0)P$, it is hard to compute $r_m P$ using $C_1$ due to the difficulty of solving the elliptic curve discrete logarithm problem. Given $U_{m,i} = r_m U_i = r_m k P_i$, $i = 1,2,...,q = n-|\mathbb{P}|$, this problem can be reduced to the $(q-1)-DHI$ problem as follows. Let $Q = \alpha r_m P$. The parameters are then rewritten $U_{m,i} = r_m U_i = \alpha^i r_m P$ as $Q_i = U_{m,i} = \alpha^{i-1} Q$, $i = 1,2,...,q$. This implies that if an adversary $\mathscr{A}$ has the ability to solve the $(q-1)-DHI$ problem, he/she can compute the key $r_m P = (1 \,/\, \alpha) Q_1 = (1 \,/\, \alpha) Q$, and then successfully decrypt the ciphertext $CT$. The following theorem proves that solving the $(q-1)-DHI$ problem is as hard as the $q-GDH$ problem. □

**Theorem 3.** *If the $(t, q-1, \varepsilon)-DHI$ assumption holds in $\mathbb{G}$, the $(t, q, \varepsilon)-GDH$ assumption also holds in $\mathbb{G}$.*

*Proof.* Suppose $\mathscr{A}$ is an algorithm that has advantage $\varepsilon$ in solving the $q-GDH$ problem. An algorithm $\mathscr{B}$ is constructed that solves $(q-1)-DHI$ with the same advantage $\varepsilon$. This follows the same proof as presented in .

Algorithm $\mathscr{B}$ is given $Q, \alpha Q, \alpha^2 Q, ..., \alpha^{q-1} Q \in \mathbb{G}$ as inputs, and its purpose is to compute, $(1 \,/\, \alpha)Q \in \mathbb{G}$. Let $R = \alpha^{q-1} Q$ and $y = 1 \,/\, \alpha$. Then, the input of $\mathscr{B}$ can be re-written as $R, yR, y^2R, ..., y^{q-1}R \in \mathbb{G}$ and $\mathscr{B}$ 's goal is to output $y^qR = (1 \,/\, \alpha)Q = T$

Algorithm $\mathscr{B}$ first picks $q$ random values $r_1, r_2, ..., r_q \in Z_p$. After that it runs the algorithm $\mathscr{A}$ and simulates the oracle $\mathscr{O}_{R,a}$ for $\mathscr{A}$. $\mathscr{B}$ will use the the vector $\mathbf{a} = (y + r_1 +, ..., y + r_q)$. Note that $\mathscr{B}$ does not know $\mathbf{a}$ explicitly since $\mathscr{B}$ does not have $y = 1/\alpha$. When $\mathscr{A}$ issues a query for $\mathscr{O}_{R,a}(S)$ for some strict subset $S \subset \{1, 2, ..., q\}$, the algorithm $\mathscr{B}$ responds as follows:

- Define the polynomial $f(x) = \prod_{i \in S}(x + r_i)$ and expand the terms to obtain $f(x) = \sum_{i=0}^{|S|} f_i x^i$.
- Compute $Y = \sum_{i=0}^{|S|}(f_i y^i R) = f(y)R$. Since $|S| \leq q$, all the values $y^i R$ in the sum are known to $\mathscr{B}$.
- By construction we know that $Y = (\prod_{i \in S}(y + r_i))R$. Algorithm $\mathscr{B}$ responds by setting $\mathscr{O}_{R,a}(S) = Y$.

The responses to all the oracle queries of the adversary are consistent with the hidden vector $\mathbf{a} = (y + r_1, ..., y + r_q)$. $\mathscr{A}$ will then output $Z = (\prod_{i=1}^{q}(y + r_i))R$. Define the polynomial $f(x) = \prod_{i=1}^{q}(x + r_i)$ and expand the terms to get $f(x) = x^q + \sum_{i=0}^{q-1} f_i x^i$. To conclude, $\mathscr{B}$ outputs $T = Z - \sum_{i=0}^{q-1} f_i y^i R = y^q R$ which is the required value. □

**Remark.** *From the above discussion, the proposed scheme is collision resistant against secret keys. As a result, computing the key $k_m = r_m P$ from a ciphertext $CT$ corresponding to the access policy $\mathbb{P}$ without a valid user secret key $SK_{\mathbb{A}_K}$ is as hard as the $q - GDH$ problem. This implies that given $\{U_{m,1}, U_{m,2}, ..., U_{m,q}, C_1\}$, where $q = n - |\mathbb{P}|$, $T \in \mathbb{G}$, the $q - GDH$ problem reduces to the $(q-1) - DHI$ problem, and then decides whether $T$ is equal to $r_m P$ or a random element in $\mathbb{G}$. But as the $q - GDH$ problem is hard to solve so would be $(q-1) - DHI$. Hence, an adversary cannot derive $r_m P$ from $C_1$.*

In conclusion, if Theorems 1, 2 and 3 hold then Fog-CPS scheme is $(t, q_s, q_d, \varepsilon)$-selectively secure if the $q - GDH$ problem is $(\acute{t}, \acute{\varepsilon})$-hard, where $\acute{t} = t + \mathscr{O}(q_s(t_{inv} + nt_{mul}) + q_{H_1} nt_{em}), \acute{\varepsilon} = \varepsilon - \frac{q_{H_2}}{p}$, $n = |\mathbb{A}|$, $q = n - |\mathbb{P}|$, and $t_{inv}$, $t_{mul}$ and $t_{em}$ denote the average time required for group inverse, multiplication and point multiplication operations, respectively, and $q_{H_1}$, $q_{H_2}$ denote the number of queries made to the random oracles $H_1$ and $H_2$, respectively.

### 3.5.3 Network Devices Compromise Analysis

Having discussed the difficult of generating and/or guessing the secret keys. The impact of the compromise of Fog-CPS entities on the proposed security scheme is discussed.

#### Compromise of FA

The compromise of FA can have drastic impact on the security of the Fog-CPS system. A compromised FA can reveal the partial secret keys $SK_{\mathbb{A}_S}$ of Fog-CPS entities. An adversary in possession of a partial secret key $SK_{\mathbb{A}_S}$ and the shared attribute set $\mathbb{A}_K$ can generate the corresponding final secret key $SK_{\mathbb{A}_K}$. After having generated the final secret key, the adversary can also change the attributes agreed with FA and subsequently further compromise the network. However, if the adversary is not aware of the shared attribute set then it cannot generate the final secret key. Moreover, as the actual encryption and decryption is performed using final public and secret keys $PK_{\mathbb{A}_K}/SK_{\mathbb{A}_K}$ meaning that the communication between the CPS devices is still secure.

#### Compromise of CPS Device and Fog Nodes (Leakage of Final Secret Key)

The compromise of CPS devices and fog nodes will only leak their own secret keys. The compromise of one set of secret keys does not risk the messages encrypted under different shared attributes therefore keys. Henceforth, legitimate CPS devices can still communicate securely.

## 3.6 Trust Management System

This section presents the mathematical construction of the second component of the proposed secure integrated framework i.e. TMS. But before presenting the details of TMS, a few important notions associated with trust computation are discussed. In Fog-CPS system, the TMS is required for two purposes, 1) trust computation and 2) trust distribution. As discussed in Section 3.1, in the proposed Fog-SGC system, i.e. smart power grid control system, the FA nodes are responsible for both trust computation and distribution.

### 3.6.1 What is Trust and Credibility in TMS?

#### Trust

The concept of trust has been there for a while. It is used in several disciplines namely, psychology, sociology and computers etc. However, there is no one universally agreed definition of trust and it generally depends upon the discipline and the use case under consideration. Following this, the definition of trust as perceived in the proposed secure integrated framework is discussed. The proposed TMS adopts a performance based trust computation approach.

***"Trust of an entity is based on its performance which is computed from QoS and network communication evidence."***

The rational behind choosing performance based trust is that fog nodes are similar to small clouds but with limited resources. In many existing studies (Nair and Khan, 2010) (Habib et al., 2013) (Xiaoyong et al., 2015), the trust in a cloud service is computed based on the QoS parameters and the number of positive and negative outcomes. So, following same logic, trust in this research is based on the performance of Fog-CPS entities.

#### Credibility

The literal meaning of the credibility is "the quality of being trusted and believed in". Trust and credibility are essentially the same concept. However, there is a subtle difference, trust of an entity depends upon on its performance, but the credibility of trust is guaranteed by the robustness and the dependability of the Fog-CPS system under consideration. Precisely, the credibility of the trust implies that trust computed in the proposed TMS is trustworthy even if the system is deployed in an unprotected and hostile environment.

### 3.6.2 Trust Relationship

In a trust relationship, the trustor is an entity which shares some of its assets and/properties with another entity namely trustee for the benefit of a third party. Precisely, in Fog-CPS systems, the fog nodes and CPS devices are the trustor and FA (as it is managing TMS) is the trustee. Moreover, the beneficiary can be the Fog-CPS system users and/or other entities.

### 3.6.3 Trust Notion

Existing trust models follow different notions of trust namely, subjective, objective and hybrid. However, it is underlined that in the proposed TMS, there is only a single notion of trust which is computed from objective evidence.

### 3.6.4 Trust Computation

For a dependable Fog-CPS system, it is essential that all entities should be trustworthy. It can be ensured by calculating and subsequently assigning a trust score to each of the entities i.e. fog nodes and CPS devices. In the proposed trust computation model, the trust score of an entity is computed on the basis of QoS evidence and network communication parameters.

Moreover, the trust prediction is achieved by applying a random forest regression model. To the best of this researcher's knowledge none of the existing trust models has applied random forest regression.

#### Trust of Fog Nodes

Trust gained by a fog node reflects its performance based on directly monitored QoS evidence and network communication parameters. To be more specific, the trust score of a fog node is the aggregation of trust computed from the evidence gathered by the FA and CPS devices. The FA monitors the QoS parameters and computes the trust for fog nodes. The CPS devices also monitor some network communication parameters of fog nodes. These parameters are subsequently reported to the FA which computes trust of the fog nodes.

#### Trust of CPS devices

Similar to the fog nodes, the trust scores of CPS devices are also computed. Both fog nodes and CPS device evaluate each other based on same set of parameters as listed in second column of Table 3.2. They subsequently report them to the FA node which computes trust for CPS devices.

### 3.6.5 Trust Distribution

Moreover, the second purpose of the TMS is to disseminate trust scores. Any device can look up and/or query the FA and/or TMS to acquire the trust scores of other entities.

### 3.6.6 QoS Monitoring and Service Matching

As the name suggests this module includes features for monitoring the service quality parameters and matching the services as per the given requirements. As discussed in Section 3.6, there are two types of CPS devices namely, fixed and mobile. Generally, service matching is not required in fog scenarios where fixed CPS devices are deployed because the interactions among devices are already established. The entities function as per the requirements of specific application scenario. However to generalize the TMS, a service matching feature is added such that the proposed TMS can also be applied to fog scenarios with mobile CPS devices. For service matching, a set of requirements is taken as input and subsequently the resource matching is carried out on the available fog nodes. FA selects highly trusted fog nodes based on their trust values.

As mentioned in section 3.6.1 that proposed TMS adopts a performance based trust computation approach. So for evaluating the performance, a number of QoS and network communication parameters of fog nodes are monitored. Additionally, for trusted service matching, it is essential to monitor the real-time service parameters of fog nodes. In the interactive process, this module dynamically monitors the service parameters and is responsible for getting run-time service data. To be more specific, the FA monitors four kinds of parameters (see Table 3.2), namely fog node specification, average resource usage, average response time and average task success ratio. The fog node specification profile includes CPU frequency (GHz percentage), memory size (GB), hard disk capacity (TB). The average resource usage information consists of the current CPU utilization rate, current memory utilization rate, current hard disk utilization rate, and current bandwidth utilization rate. The fog node profile specification informs about the overall resource capacity and their average resource usage indicates their current utilization which in turn helps in determining how many service requests can be entertained.

### 3.6.7 Objective Trust Computation

As trust is dynamic and changes over time, so the proposed TMS is designed by taking this into consideration. Trust computation of the Fog-CPS entities takes place over a time duration $\tau$ meaning that trust is a function of time. Time duration $\tau = \{1,2,\ldots,t,\ldots,\acute{t}\}$ is divided into several discrete time instants $t$.

Let $fog$ is a fog node and $cps$ is a CPS device requesting a service at time instant $t$. Once the service request has been fulfilled by the fog node $fog$, the FA monitors and/or calculates its QoS parameters and $cps$ device also sends its parameters. The FA is equipped with monitors which can pull the real-time service parameters listed in Table 3.2. CPU frequency, memory size and disk utilization are static so they are not monitored. The FA pulls up current CPU, memory and hard disk utilization rates and subsequently takes the mean of utilization rates over time duration $[t-1,t]$ to capture all the variations. Same goes with the task success ratio and response time. FA computes the average response time by taking the mean of service response times over time duration $[t-1,t]$. Similarly, the average task success ratio of $fog$ is also calculated. The task success ratio is calculated by dividing the number of service requests that has been assigned to a fog node and the number of requests which were successfully completed in a given time instant $t$. For instance, if a node has been assigned ten service requests and it delivered 7 then the the task success ratio is 0.7. Subsequently, the task success ratio over time duration $[t-1,t]$ is averaged.

Once all the QoS parameters are monitored and/or calculated, random forest regression (Criminisi et al., 2012) is employed to predict the objective trust of $fog$ at time instant $t$. In

prediction problems, the regression models are trained over a substantial number of samples in order to improve accuracy. So for objective trust computation, the first task was to generate the trust labels corresponding to a set of service parameters. The trust label is assigned based on average task success ratio and service quality.

Similar to SLA which is defined to evaluate the service quality provided by the cloud service providers, the FA defines a set of QoS parameters that indicates the expected QoS that would be provided by each fog node based on its device specification (i.e. trust parameters in this case). So for trust assignment, the average task success ratio and the service parameters are compared with the predefined set of QoS parameters. Trust gets a high value if the average task success ratio is high and the fog node fulfilled the service request by maintaining the acceptable service quality and vice versa.

### Initial Trust Label

Each newly added fog node and CPS device is assigned a trust value of 0.5. Same applies to the entities who did not interact with any other entity before. However, as the network operates, trust changes based on the performance of the fog node which is determined by average task success ratio and service quality.

Following this, the random forest regression is executed to predict the objective trust at a given time instant $t$ based on the service parameter features. As the QoS features are high dimensional (i.e. each sample has multiple features to consider). At each node in a tree, the optimization happens by selecting one feature randomly and optimizing for it. This process is repeated multiple times until the best feature is found and subsequently split at that node. Once this happens, the data is split based on that feature and each split is passed to the two nodes below. As the training finishes, the trust labels for each set of QoS parameters calculated at different discrete time instants are obtained. Trust label at time instant $t$ is denoted by $T_t$ However, as mentioned above, the trust evaluation is over time duration $\tau$. So, the objective trust $T_\tau^{fa\rightarrow fog}$ is calculated using Equation (3.27):

$$T_\tau^{fa\rightarrow fog} = \sum_{t=1}^{\tau}(T_t(fog)w_t(fog)), \tag{3.27}$$

$w = \{w_1(fog), w_2(fog), ..., w_\tau(fog)\}$, and $\sum_{t=1}^{\tau} w_t(fog) = 1$. $w_t(fog) \in [0,1]$ is the weight assigned to each objective trust $T_t(fog)$ at time instant $t$ and is given by Equation (3.28).

$$w_t(fog) = \frac{(1-(t+1)^{-1})}{\sum_{t=1}^{\tau}(1-(t+1)^{-1})} \tag{3.28}$$

Table 3.2 Fog Node Trust Parameters

| **Fog Assist** | **CPS Device** |
|---|---|
| *FA Trust Parameters* | *CPS Trust Parameters* |
| CPU frequency | Energy Consumption |
| Memory size | Response Time |
| Hard disk capacity | Bandwidth |
| Current CPU utilization rate | |
| Current memory utilization rate | |
| Current hard disk utilization rate | |
| Average response time | |
| Average task success ratio | |

$w$ is a time-based attenuation function which assigns more weight to $T_t(fog)$ computed at recent time instants.

### 3.6.8 Trust Parameter Monitoring

A "Trust Parameter Monitoring" module is installed in each CPS device and will enable them to quantify the utilization of energy, bandwidth and response time when communicating with a fog node.

### 3.6.9 CPS Device Parameter Monitoring

Similar to CPS devices, the fog nodes also monitor a few parameters for each CPS device which is connected to it. Both fog nodes and CPS devices evaluate each other on same set of parameters. The fog nodes subsequently report them to FA which computes the trust for CPS devices.

### 3.6.10 Data Anomalies Detection

After the multi-dimensional parameters (table 3.2) related to communication features are reported by CPS devices and fog nodes. As the energy consumption, response time and bandwidth of CPS devices are quantifiable and must have certain range (i.e. minimum and maximum values). For example, the energy consumption of a raspberry PI 3 is 1.2 joules per second or watt. The energy consumption cannot exceed beyond the minimum and maximum values of the range. Same is the case with other parameter values. The data

anomalies are identified by comparing the parameter values with the predefined range of each of these parameters. If a parameter value falls within the range, then it is considered for trust computation otherwise not.

### 3.6.11 CPS Device Trust Computation

This module computes the trust of CPS devices for fog nodes. Similar to the objective trust, the CPS device trust based on a set of parameters is predicted by the random forest regression model. Precisely, regression evaluates the relationship between parameters and trust. A trust value is predicted for each observation. As the CPS devices can send multiple parameter reports in a given discrete time instant $t$. So, the CPS device trust $T_{\tau}^{cps \to fog}$ is the mean of all trust values predicted by regression model based on the parameter reports sent during a time duration $\tau$. Finally, the CPS device trust $T_{\tau}^{cps \to fog}$ for a fog node $fog$ is computed using Equation 3.29.

$$T_{\tau}^{cps \to fog} = \frac{\sum_{i=1}^{r} c(i, fog)}{r} \tag{3.29}$$

where $c(i, fog)$ is the trust computed from the $i$th report sent by the CPS devices provisioning services from a fog node $fog$ and $r$ is the total number of parameter reports. $c(i, fog)$ is computed using random forest regression. The credibility of trust is evaluated based on the model discussed in the following subsection.

### 3.6.12 Trust Credibility Evaluation

As discussed in section 1.2, the Fog-CPS systems can be deployed in open and unprotected locations and are therefore at the risk of compromise. Moreover, such distributed systems can also be subject to collusion, on-off, bad-mouthing, and self-promotion attacks. Compromised entities might try to change the trust of other nodes by reporting false parameters. For instance, in collusion attacks, the attackers can either work alone or in coalitions to increase/decrease the trust of Fog-CPS entities.

The key idea behind the credibility model is to capture the change in trust over a time duration. The change can occur in both legitimate and hostile environments, and sophisticated attackers can also slightly change the trust parameter to affect the trust of other nodes. Solving this problem is not straightforward because on the one hand, it is not easy to predetermine the number of compromised entities and on the other hand, it is essential to minimize the impact of malicious attackers. Keeping these constraints in mind, a trust

credibility evaluation model is designed to adjust the trust of Fog-CPS entities in three cases whereby the CPS devices, fog nodes and FA could be compromised.

**Case 1 - Compromise of CPS Devices:** Compromised CPS devices may try to change $T^{cps \rightarrow fog}$ by reporting false parameters. The proposed credibility model and data anomalies detection modules can handle these discrepancies.
**Case 2 – Compromise of Fog Nodes:** Compromised fog nodes can report false parameters to change the $T^{fog \rightarrow cps}$. Similar to case 1, this problem is redressed by monitoring the rate of change of trust and subsequently adjusting it based on trust computed in previous time instances.
**Case 3 – Compromise of FA:** As the TMS model is maintained by the FA node so its compromise can lead to following problems:

1. *Inaccurate Computation of $T^{fog}$, $T^{cps \rightarrow fog}$ and $T^{fog \rightarrow cps}$:* The FA computes the trust of Fog-CPS entities and then store it in the Trust Evidence Database which is publicaly accessible. So, if an entity finds a discrepancy in its trust score, it can request the recomputation of trust based on the current values of parameters. If its request is not entertained then it can inform all involved fog nodes and CPS devices.

2. *Tampering the QoS and CPS device Parameters:* A compromised FA node can also change the parameter values reported by the Fog-CPS devices. This can be addressed by the making the "Trust Evidence Database" accessible to the relevant entities. The fog nodes and CPS devices can verify and/or compare their reported set of parameters to those stored in the Database. The discrepancy could be reported back to the FA node and the involved entities.

To address all these cases of compromise, trust credibility evaluation is applied in all computations i.e. $T^{fog}$, $T^{cps \rightarrow fog}$ and $T^{fog \rightarrow cps}$ by default. Hence, any "large" differences will be adjusted. Precisely, when new CPS devices and fog nodes are taking part in the network their trust value is expected to be 0.5. While the network operates trust values will be increased or decreased. As the trust credibility model is applied in all trust computations, so a generalized notion of trust $T$ is introduced but it should be considered in all computations. Trust credibility evaluation model analyses the change in $T$ during two consecutive time duration $[\tau-1, \tau]$ and subsequently recomputes the trust in recent time duration $\tau$ using Eq. (3.30):

$$\acute{T}_{\tau} = T_{\tau-1} \pm \acute{\sigma} T_{\tau}, \tag{3.30}$$

where $\acute{\sigma}$ is the standard deviation in $T$ over a time duration $\tau$. The standard deviation $\acute{\sigma}$ informs about the spread of the possible values of trust. $\sigma$ is computed by Eq. (3.31):

$$\acute{\sigma} = \sqrt{\frac{\sum(T-\mu)^2}{\tau}} \tag{3.31}$$

where $\mu$ is the sample mean of trust $T$ values calculated during time duration $\tau$. The standard deviation should be taken/considered for every newly calculated trust value over a time duration $\tau$. If the trust $T$ in recent time duration $\tau$ is less than the previous time duration $\tau$–1 and the difference is greater than $\sigma$ then the $T$ in $\tau$ is increased otherwise it is decreased.

### 3.6.13 Fog Node Trust Computation

Having discussed objective $T_\tau^{fa\to fog}$ and CPS $T_\tau^{cps\to fog}$ trust modules, the next task is to aggregate them to compute trust of a fog node $T_\tau^{fog}$ and to assign weights to objective and CPS trust. Through weight assignment, it is easy to define the proportion of CPS and objective trust in trust evaluation of a fog node. $T_\tau^{fog}$ is calculated as follows:

$$T_\tau^{fog} = \delta \times T_\tau^{cps\to fog} + (1-\delta) \times T_\tau^{fa\to fog} \tag{3.32}$$

where $\delta$ is the weight of $T_\tau^{cps\to fog}$, and, $(1-\delta)$ is the weight of $T_\tau^{fa\to fog}$. The value of $\delta$ does not have a fixed value but as the credibility model is already monitoring the rate of change of trust and readjusting it. So, both the $T^{fog}$ and $T^{cps->fog}$ are assigned equal weights. But it could be changed as the network operates and with increasing number of malicious devices. If the value of $\delta$ is set to 1, the weight of $T_\tau^{fa\to fog}$ becomes 0, and Equation 3.32 will only consider CPS trust. However, many studies (Kim et al., 2010; Tian et al., 2010; Xiaoyong and Yuehua, 2011) show that objective trust $T^{fa\to fog}$ is a helpful component in building a dependable trust relationship. When the system is highly dynamic and most CPS devices are malicious, the objective trust $T_\tau^{fa\to fog}$ should be set with a high weight. Intuitively, the value of $T_\tau^{cps\to fog}$ calculated above should have a higher weight if the number of rating CPS devices is higher.

### 3.6.14 Trust Computation for CPS Devices

Similar to fog nodes, the trust of CPS devices $T^{fog\to cps}$ is also computed. Both fog nodes and CPS devices assess each other on the same set of communication parameters i.e. energy consumption, bandwidth, and response time. Each fog node quantifies the above parameters

for all CPS devices and subsequently reports them to the FA. Trust of cps devices $T^{fog \to cps}$ is also computed using Eq. (3.29).

### 3.6.15 Trust Evidence Database

FA stores the trust values of each fog node and CPS device in the trust evidence database. Any device can look up and/or query FA and acquire the trust scores of other entities and based on this makes a decision to collaborate. Through this database, trust scores can be easily disseminated across the entire Fog-CPS system operating in a given region.

Table 3.3 Attributes of Fog-CPS entities and Trust Integration

| **IDs** | **Access Control Rights** | **Trust** | **Trust Attribute** |
|---|---|---|---|
| $cps_{01}$ | None | <0.3 | 000 |
| $cps_{02}$ | Read | >= 0.3 - 0.4 | 001 |
| $cps_{03}$ | Write | > 0.4 - 0.5 | 010 |
| $fog_{01}$ | Delete | > 0.5 - 0.6 | 011 |
| $fog_{02}$ | Read and Execute | > 0.6 - 0.7 | 100 |
| $fog_{03}$ | Modify (Permissions) | > 0.7 - 0.8 | 101 |
| $fog_{04}$ | Special Permisssion | > 0.8 - 0.9 | 110 |
| $fog_{05}$ | All | > 0.9 - 1 | 111 |

## 3.7 Integration of SC and TMS

The SC and TMS components have already discussed in preceding sections. In this section, their integration is discussed. The underlying idea behind integration is to embed trust as an attribute in the access control policy $\mathbb{P}$ in SC component. Table 3.3 describes a few attributes which can be used for identity and access management in SC. The columns **IDs** and **Access Control Rights** list the identities and the access rights of fog nodes and CPS devices. Likewise, the columns **Trust** and **Trust Attribute** describe the trust and its corresponding representation required for the access rights granted to a Fog-CPS entity. It can be observed that change in trust is triggering a change in access rights, for instance, an increase/decrease in trust is resulting in privilege escalation/deescalation respectively.

Following this, the integration is explained based on the attribute set and access policy structures defined in section 3.3.4. For example, if $\mathbb{A} = \{A_1, A_2, A_3, A_4, A_5, A_6\}$ is the attribute set of a CPS device $cps_{01}$ and it has shared $\mathbb{A}_K = \{A_1, A_2, A_3\}$ with a fog node $fog_{01}$ then its attribute string is represented as $\mathbb{S}_K = 111000$. With regard to embedding trust in $\mathbb{S}_K$ and

$\mathbb{S}_P$, its binary representation (i.e. column **Trust Attribute** in Table 3.3) corresponding to an access right is added to the device attribute set and access policy strings. Let us suppose $A_6$ attribute now represents the access rights of a CPS device $cps_{01}$ which can "Write" (i.e. 010) then its $n$-bit string would be denoted as $\mathbb{S}_K = 11100010$. If $\mathbb{P}$ is defined over $A_1, A_2, A_3, A_6$ and $A_6 = 001$ which is "Read" access (i.e. 001) then it is denoted as $\mathbb{S}_P = 11100001$. It is maintained that a CPS device with attribute set $\mathbb{A}_K$ fulfills the access policy $\mathbb{P}$, if and only if $\mathbb{S}_P \subseteq \mathbb{S}_K$. In other words, the device access rights which are granted based on its trust value must be greater than or equal to the access policy otherwise the access is denied. A simplest example of access denial could be the inability to decrypt the ciphertext $CT$.

## 3.8 Concluding Remarks

Fog-CPS systems carry sensitive information (i.e. power consumption patterns, identities, health monitoring data, and mobility traces etc). The leakage of such sensitive information can risk the security and privacy of the end users. Additionally, Fog-CPS systems face several trust challenges due to support for features namely low latency, mobility, location awareness and decentralized architecture. To address these challenges, a secure integrated framework for Fog-CPS systems is proposed. The framework is designed after a thorough investigation of these systems. Various dimensions (i.e. security, trustworthiness and service orchestration) of Fog-CPS systems are studied to find the vulnerabilities and threats faced by such complex and inherently heterogeneous systems. After identifying the security and trust challenges, efforts were made to find a solution. However, soon it had been clear that Fog-CPS systems require an integrated approach which addresses these issues simultaneously. As the limitations and/or absence of one solution can be exploited by malicious attackers to disrupt these systems and impact their availability.

The SC component ensures the security by achieving data confidentiality, authentication and access control through a lightweight ABE scheme based on elliptic curves. The identity management and access control management subcomponents guarantee that fog nodes and CPS devices are authenticated and authorized. With this Fog-CPS scheme, the sensitive attributes are associated with the secret keys and only authorized entities can access the private information. Linking the identifying and sensitive attributes with the keys and subsequently using them to encrypt the communication between collaborating entities guarantees protection against *data privacy*, *unauthorized secondary usage*, *data disclosure* and *data correlation* attacks. Moreover, each Fog-SGC entity registers with the FA based on a set of attributes which in turn safeguards against *identity impersonation*, *Sybil* and *forgery* attacks. The CP-ABE style encryption with access policy guarantees the enforcement of robust *identity and*

*access management*. Furthermore, due to encrypted communication between collaborating entities, *eavesdropping* is of little use. Lastly, the approach of dividing the attributes into secret and shared attributes empowers the CPS devices and users to have control over their sensitive data. Hence the major challenge of *lack of user control* in Fog-CPS systems is addressed.

However, an encryption scheme alone cannot guarantee the dependable behaviour of the Fog-CPS entities. To address this problem, a TMS is proposed which computes the trust for fog nodes and CPS devices in a Fog-CPS system. The TMS guarantees the dependability of Fog-CPS entities by computing their trust based on QoS parameters and other performance indicators by employing the random forest regression model. Moreover, considering the possible deployment of Fog-CPS systems in hostile and unprotected environments and the compromise of the CPS devices and fog nodes, the credibility of trust is evaluated. A credibility evaluation model is designed to countermeasure the malicious behaviour (i.e. collusion, Sybil, self-promotion and bad-mouthing) of compromised entities. With the proposed TMS, each entity can find the trust score of other entities and based on this makes a decision to collaborate.

The integration of SC and TMS is achieved by embedding trust as an attribute in access control policy in the proposed Fog-CPS scheme. Access control rights are granted based on the trust of a fog node and CPS device. Precisely, when new CPS devices and fog nodes are taking part in the network their trust value is expected to be 0.5. While the network operates trust values will be increased or decreased and subsequently trigger change in access control rights granted to Fog-CPS entities. It is believed that proposed secure integrated framework can address the security and trust challenges faced by the Fog-CPS systems.

## 3.9 Summary of the Chapter

This chapter presented the proposed secure integrated framework. Section 3.1 proposed a generalized fog-enabled smart power grid control system (Fog-SGC). Security and privacy issues faced by Fog-CPS systems were further discussed in Section 3.1.3. Additionally, the challenges in devising secure solutions for such systems were outlined in Section 3.1.5. The security requirements of a generalized Fog-CPS system were also underlined 3.1.6.

The proposed integrated framework is presented in Section 3.2. Following that the mathematical constructions of both components i.e. SC and TMS are added. At the beginning, the secure integrated framework is described in Section 3.2. The Fog-CPS scheme and its mathematical constructions i.e. algorithms are described in Sections 3.3 and 3.4. The security

analysis is presented in Section 3.5. The integration of the trust and encryption scheme is discussed in Section 3.7. Lastly, the TMS is presented in Section 3.6.

# Chapter 4

# Experimental Evaluation

This chapter presents the experimental results of the proposed secure integrated framework. The proposed framework consists of two fundamental components, namely security component (SC) and trust management system (TMS). Both of these components are experimentally evaluated and their results are presented in the following sections. First the results of the encryption scheme are presented and then the results of TMS are discussed.

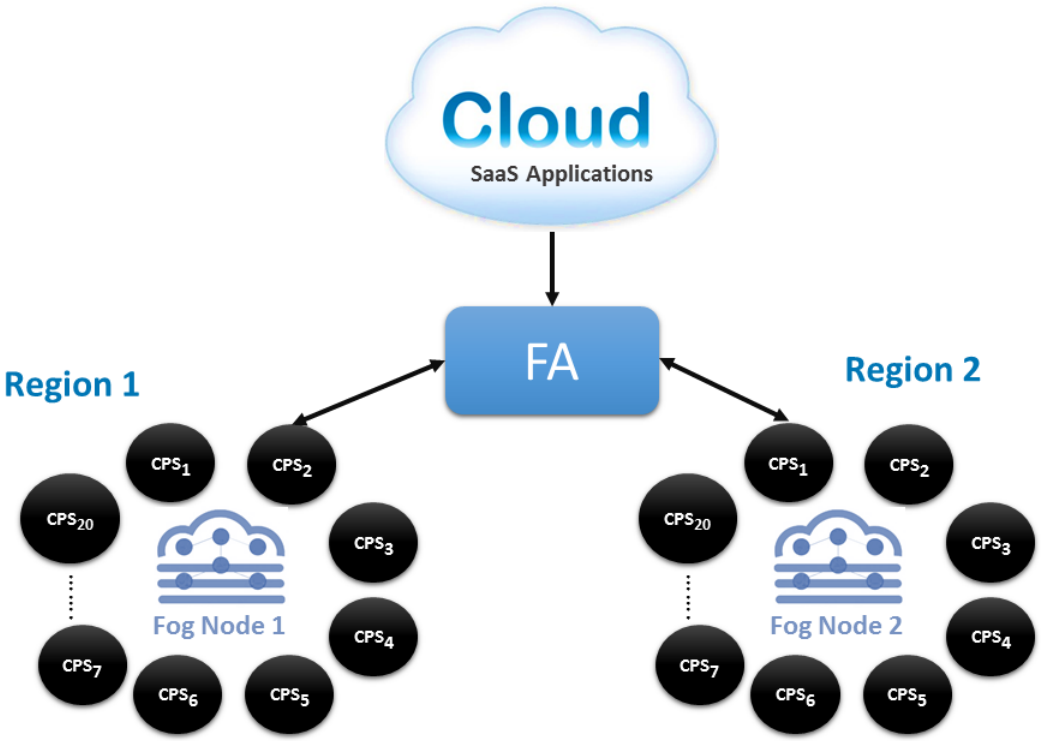


Fig. 4.1 Experimental Set-up

## 4.1 Experimental Evaluation of SC

The experimental setup for evaluating the proposed secure integrated framework is shown in Figure 4.1. It can be observed that the Fog-CPS system has three types of entities, namely CPS devices, fog nodes and cloud service provider. Fog nodes and cloud provider can support computationally expensive operations. On the contrary, CPS devices have limited resources. To evaluate the performance of the proposed scheme, its algorithmic efficiency in terms of time and memory complexity is measured

### 4.1.1 System Configurations

For benchmarking the time complexity, two sets of experiments are conducted to demonstrate the effectiveness of the proposed scheme on both resource limited CPS devices and resourceful fog nodes. In the ***first experiment***, the scheme is evaluated on a Raspberry Pi 3 model B+ (CPS devices). It has Quad Core 1.2GHz, 64 bit CPU, 1 GB of RAM, a wireless LAN and Bluetooth Low Energy (BLE) on board, 100 Base Ethernet, 40-pin extended GPIO, 4 USB ports, HDMI and micro SD port. In the ***second experiment***, it is executed on a virtual machine running Ubuntu R16.04 with Python 3.6.4. on Intel (R) Core(TM) i5-4310U CPU@2.000GHz with 8.0 GB RAM (Fog Nodes).

All ABE schemes, including the one described here, are based on a selective security model in which the adversary submits the access policy to challenger before seeing the public/secret key pair. The results were also compared with ECC-Auth-18 (Mahmood et al., 2018) which proposes an authentication scheme with elliptic curves for smart grid.

### 4.1.2 Implementation and Evaluation

Fog-CPS scheme is compared with five other ABE schemes ECC-CPABE (Odelu and Das, 2016), CPABE14 (Guo et al., 2014), CPABE14 (Guo et al., 2014), PP-CPABE15 (Zhou et al., 2015), CPABE11 (Chen et al., 2011) and COM-CPABE14 (Shota Yamada and Kunihiro, 2014). All security schemes, including this, are based on a selective security model. The schemes are implemented in Charm (Akinyele et al., 2013); a framework developed to facilitate the rapid prototyping of cryptographic schemes and protocols. It is based on the Python programming language. However, the routines that implement the dominant group operations use the PBC library (Lynn, 2007b) (written natively in C) and the time overhead imposed by the use of Python is usually less than 1%.

It is noted that the proposed scheme is not based on bilinear elliptic curves and can be implemented on any elliptic curve. However, in order to compare the scheme with existing

ABE schemes which are based on bilinear maps, it is implemented on bilinear curves i.e. MNT159 and SS512. On other curves namely, prime192v1 and secp224r1, the memory overhead would be less. The proposed scheme and two other (Guo et al., 2014), (Odelu and Das, 2016) are tested on non super-singular asymmetric bilinear curve (i.e. MNT159). Whilst three of the schemes (Chen et al., 2011; Shota Yamada and Kunihiro, 2014; Zhou et al., 2015) have been tested on super-singular SS512 curve. Both SS512 and MNT159 curves provide 80-bit security.

### Timing Results

The execution times of all algorithms are benchmarked to compare the efficiency of different schemes. In existing schemes ECC-CPABE (Odelu and Das, 2016), CPABE14 (Guo et al., 2014), PP-CPABE15 (Zhou et al., 2015), CPABE11 (Chen et al., 2011), and COM-CPABE14 (Shota Yamada and Kunihiro, 2014), the *Setup* and *KeyGen* algorithms are separate. But, since there is no *Setup* in the Fog-CPS scheme, the execution time of both these algorithms is added and compared with the timing of final public and secret key generation. Precisely, the final key pair generation timing of this scheme is the sum of execution times of Algorithms 2 and 3.

For *Setup* and *KeyGen*, three different size of attribute universe $\mathbf{U}$ and user attribute sets $\mathbf{A}$ are considered. Precisely, an attribute universe $\mathbb{U}$ of 10, 20 and 30 attributes has been implemented for measuring the timing of *Setup* Algorithm. Likewise, for secret key generation, a user attribute set $\mathbb{A}$ (shared attribute set $\mathbb{A}_K$ for final secret key generation in this case ) of 5, 15 and 25 attributes is taken into consideration.

Moreover, two types of benchmarks are set for measuring the times for encryption and decryption, 1) 1 KiloByte (1 KB), and 2) 1 MegaByte (1 MB). These two low size messages are used because CPS devices and cloud requests are usually transmitted in low size messages. Furthermore, in encryption algorithm, an access policy $\mathbb{P}$ of constant size i.e. 5 attributes is considered. For Key Update, there are two cases with an increment and decrement of one attribute i.e. $t \pm 1$. However, in our experimental evaluation, the execution time for key update are recorded in case of $t+1$ attributes only.

### First Experiment

In this experiment, the implementations are carried out on Raspberry Pi 3B+ model. Timing results are shown in figures 4.2 - 4.9. From Figure 4.2, it can be observed that Fog-CPS scheme takes 0.02 sec for final key pair generation over 10 attributes, 0.06 sec over 20 attributes , and 0.10 sec over 30 attributes. Figures 4.6 and 4.7 show the timings of encryption,

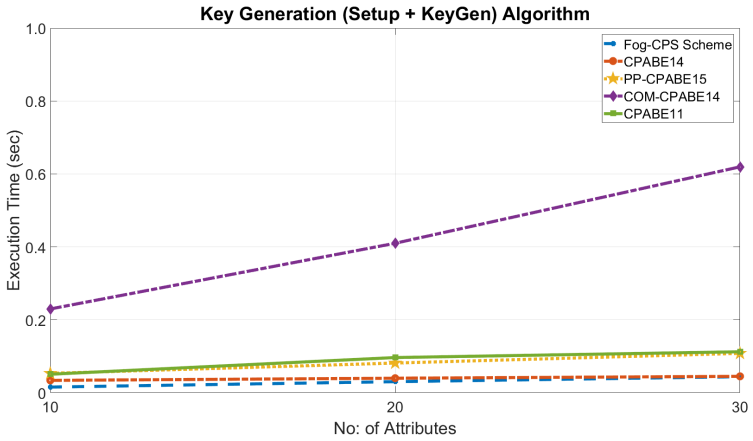


Fig. 4.2 Final Key Pair Generation Fog-CPS Scheme (Algorithms 2 & 3), and Setup + KeyGen (other schemes)

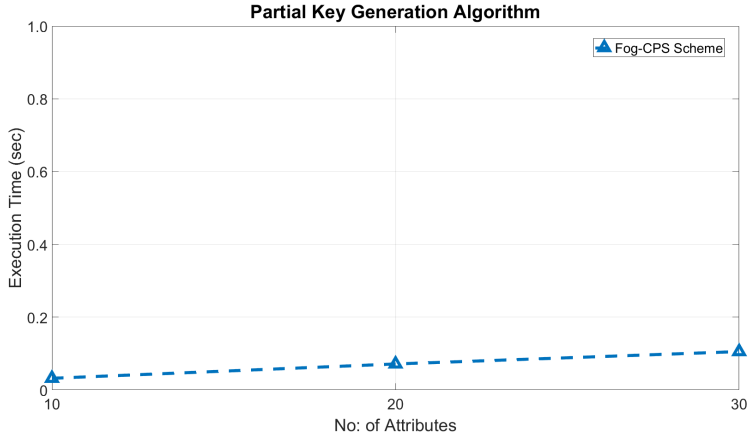


Fig. 4.3 Partial Key Pair Generation (Algorithm 1)

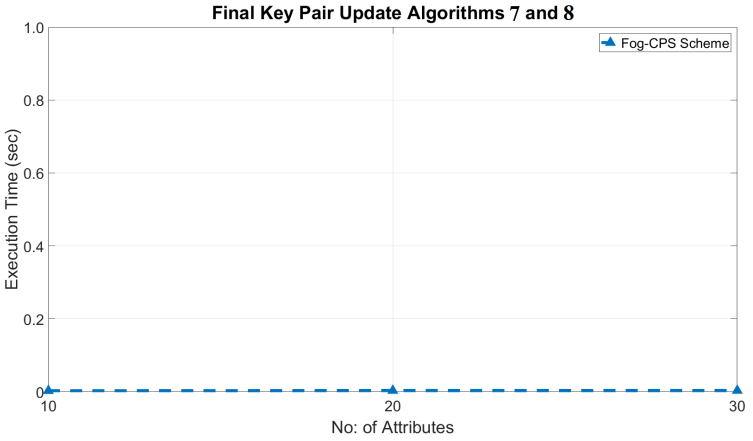


Fig. 4.4 Final Key Pair Update (Algorithms 7 & 8)

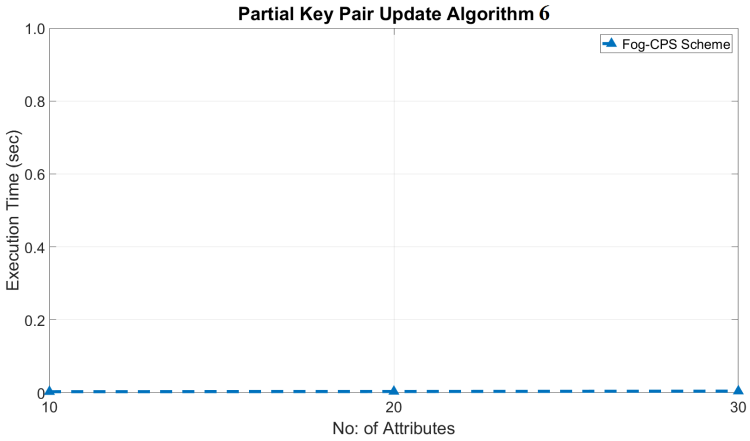


Fig. 4.5 Partial Key Pair Update ( Algorithm 6)

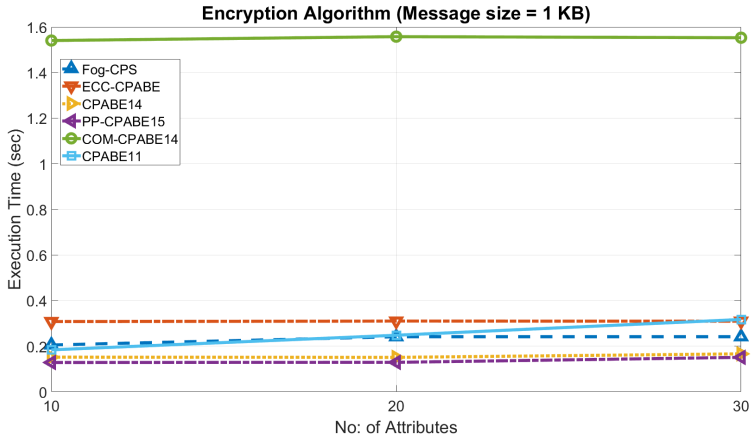


Fig. 4.6 Processing time (sec) for Encryption (Message size = 1 KB)

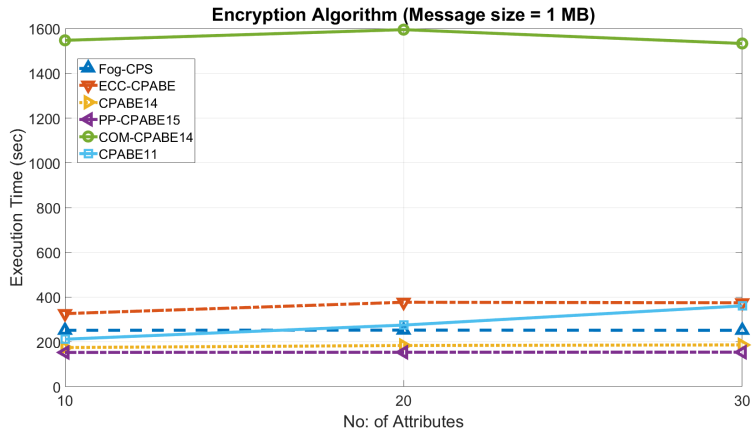


Fig. 4.7 Processing time (sec) for Encryption (Message size = 1 MB)

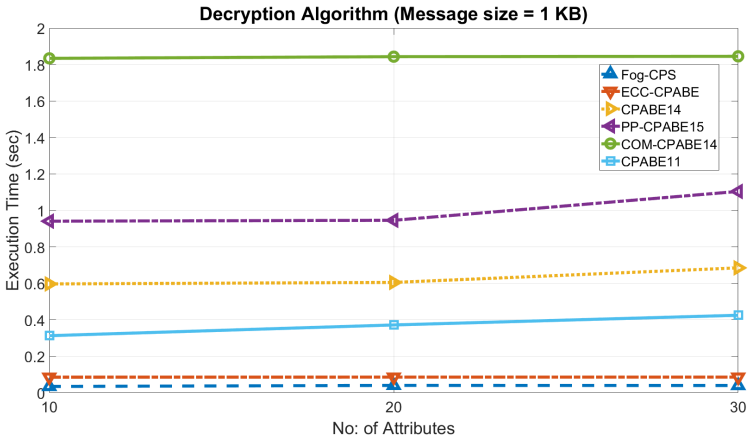


Fig. 4.8 Processing time (sec) for Decryption (Message size = 1 KB)

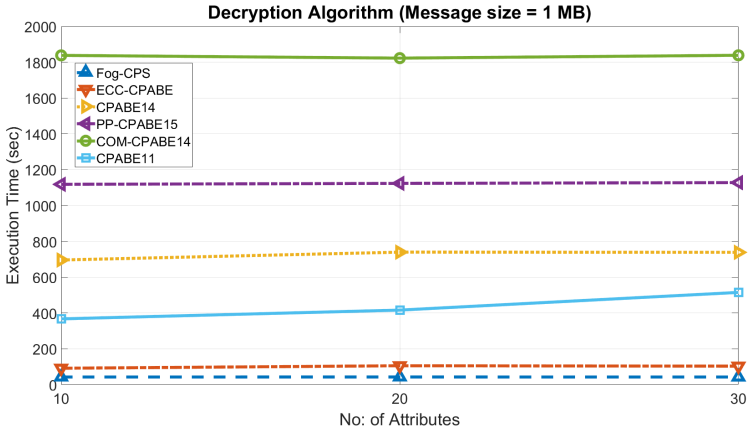


Fig. 4.9 Decryption Algorithm (Message size = 1 MB)

this scheme takes 0.20 and 252.3 secs to encrypt a message of 1KB and 1MB receptively. Similarly, for decrypting a ciphertext of 1KB and 1MB, it takes 0.03 sec and 43.1 secs as shown in figures 4.8 and 4.9 respectively. Additionally, Figure 4.3 shows the partial key pair generation timings and figures 4.4 and 4.5 show the final and partial key pair update timings of Fog-CPS scheme. It can be observed that key pair update timings of this scheme is negligible due to lightweight and efficient process.

### Second Experiment

In the second experiment, the implementations are executed on a desktop computer (configurations are mentioned in Sec. 4.1.1). Figures 4.10 - 4.17 show the timing results of final key pair generation, encryption, decryption, and key pair updates. Overall it is observed that benchmarks recorded on Raspberry Pi model 3 B+ are slower than the desktop computer.

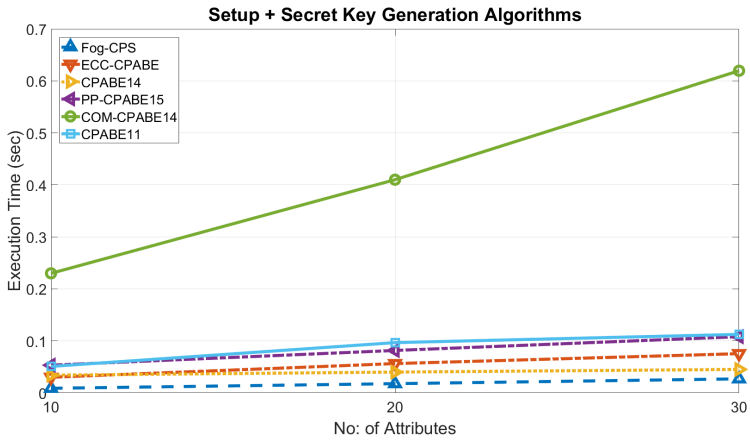


Fig. 4.10 Final Key Pair Generation Fog-CPS Scheme (Algorithms 2, 3), and Setup + KeyGen (other schemes)

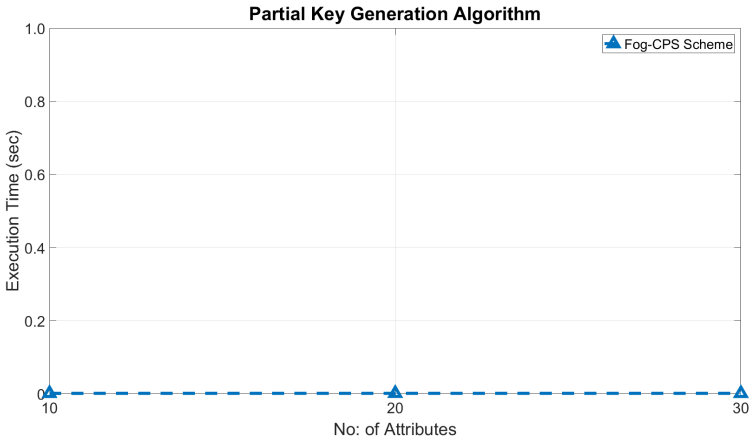


Fig. 4.11 Processing time (sec) for Partial Key Pair Generation Algorithm (1)

Comparing figures 4.2 and 4.10, it is noted that proposed scheme is slower on Raspberry Pi. But it is still fastest than the rest of the schemes as it only takes 0.009, 0.017 and 0.02 seconds for an attribute universe of 10, 20 and 30 attributes respectively. Figure 4.11 shows the execution time of partial key pair generation. Fog-CPS scheme takes 0.01, 0.018 and 0.027 seconds to generate partial key pair over 10, 20 and 30 attributes respectively. Again comparing these timings with Figure 4.3, it is clear that the timings in the ***first experiment*** are slower as it is taking more time. Figures 4.12 and 4.13 report the timings of final key pair and partial key pair update. It is clear that Algorithms 6, and 7 and 8 take same time for instance, 0.00097, 0.00098 and 0.0010 sec for 10, 20 and 30 attributes respectively. The timings are similar because only one additional exponent $(t+1)$ was applied.

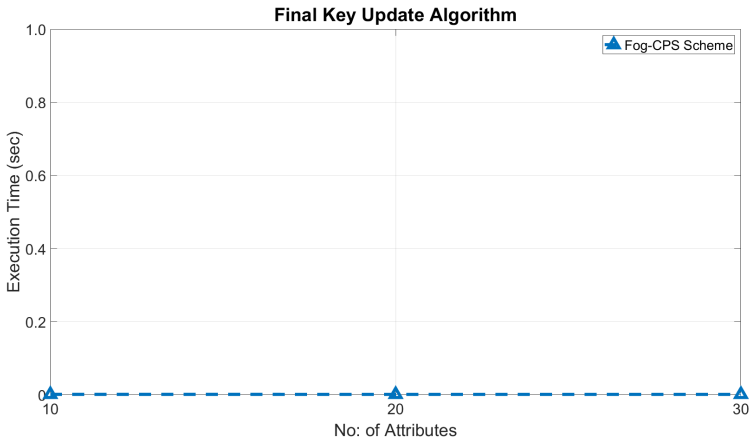


Fig. 4.12 Processing time (sec) for Final Key Update Algorithms 7 & 8)

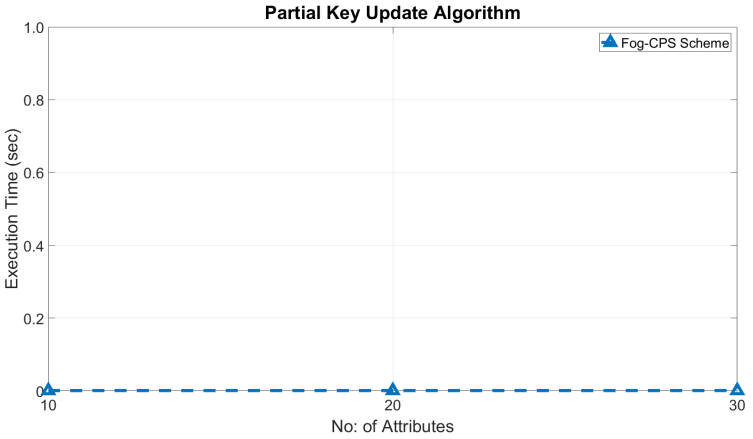


Fig. 4.13 Processing time (sec) for Partial Key Update Algorithm 6

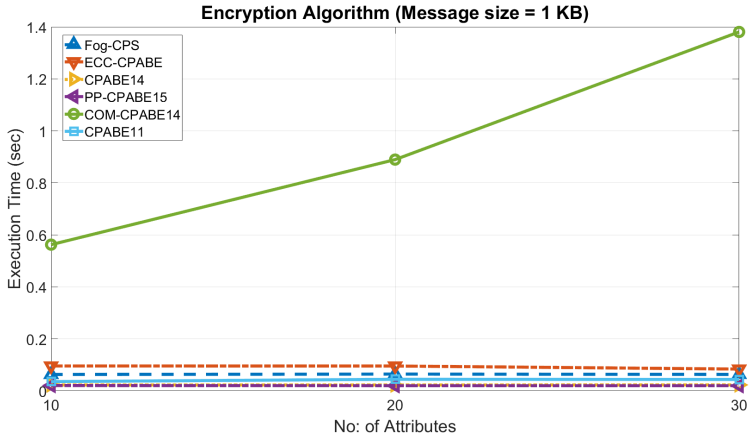


Fig. 4.14 Processing time (sec) for Encryption Algorithm 4 (1 KB Message)

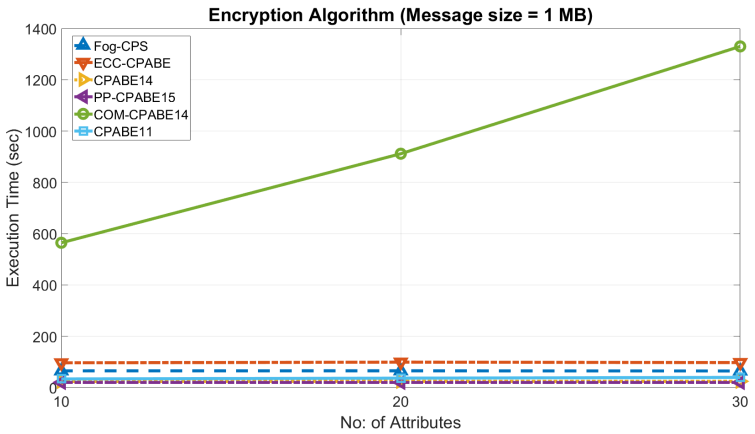


Fig. 4.15 Processing time (sec) for Encryption Algorithm 4 (1 MB Message)

The execution time for encryption of 1 KB and 1 MB messages is reported in Figures 4.14 and 4.15. The encryption timing of PP-CPABE15 and CPABE14 are almost equal and they are faster than rest of the schemes, followed by the CPABE11. Fog-CPS scheme is three times slower whereas the ECC-CPABE scheme is four times slower than the PP-CPABE15 and CPABE14. Likewise, Fog-CPS security scheme is 10 times faster than the COM-CPABE14 which is the slowest of all schemes. Comparing the timings of encryption and decryption in Figures 4.14 and 4.15, it is observed that in case of encryption, the proposed scheme is a bit slower than three of the other schemes. However, in decryption, this scheme is fastest and only takes 0.01 seconds for encryption of 1 KB message. Following the same trend as in encryption, COM-CPABE14 is the slowest of all other schemes in decryption as well. From figure 4.16, it can be observed that for decryption of 1 KB message, the execution time of ECC-CPABE is almost equal to CPABE11.

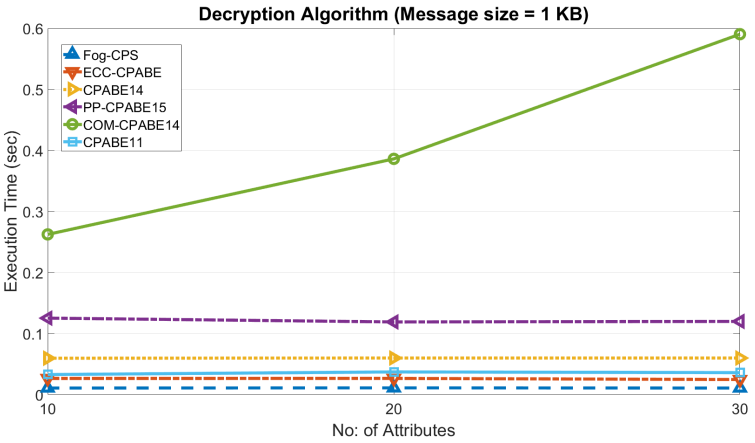


Fig. 4.16 Processing time (sec) for Decryption Algorithm 5 (1 KB Message)

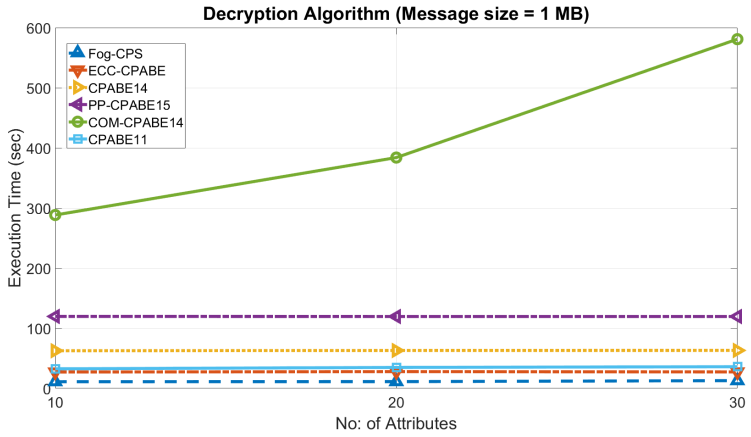


Fig. 4.17 Processing time (sec) for Decryption Algorithm 5 (1 MB Message)

In ECC-Auth-18 (Mahmood et al., 2018), there are three algorithms: initialization, registration and authentication. For the sake of comparison, the timings of these three algorithms has been added and compared with the final key generation timings of other schemes. Figure 4.18 lists the key generation, encryption, and decryption timings. ECC-Auth-18 takes 0.012 sec for key generation. The encryption/decryption of 1KB and 1MB messages using shared session key takes 0.0045 and 3.72 sec respectively. ECC-Auth-18 is the fastest followed by the scheme proposed here compared to all other schemes.

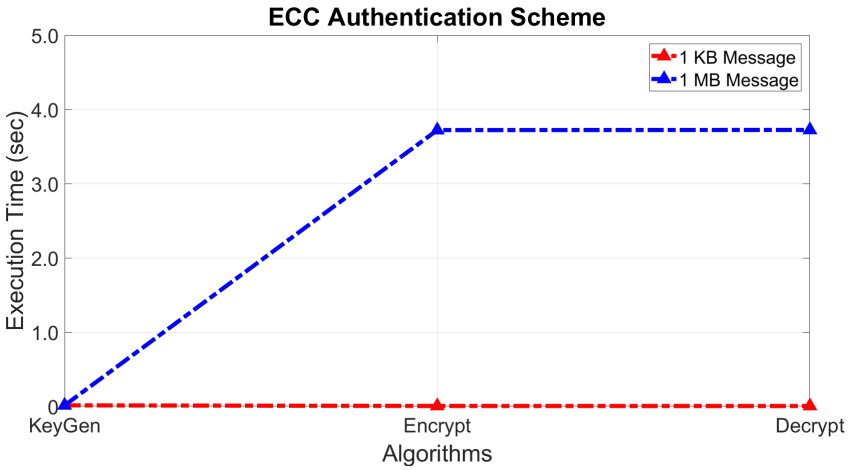


Fig. 4.18 Processing time (sec) of ECC-Auth-18 (Mahmood et al., 2018) Scheme

### Operations Results

To compare the arithmetic efficiency of the proposed scheme, the number of addition, multiplication, division and exponentiation operations (see Figures 4.19 - 4.22) of all the

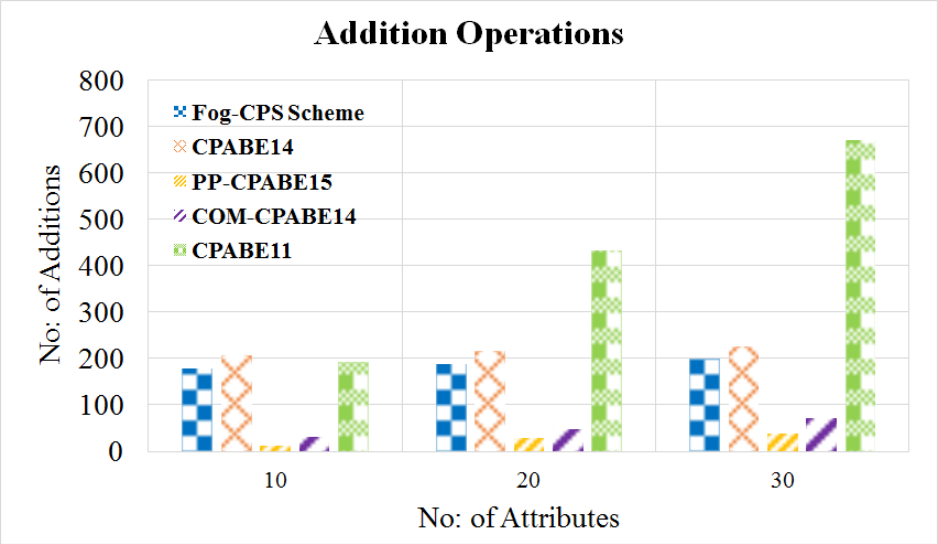


Fig. 4.19 No: of Addition Operations

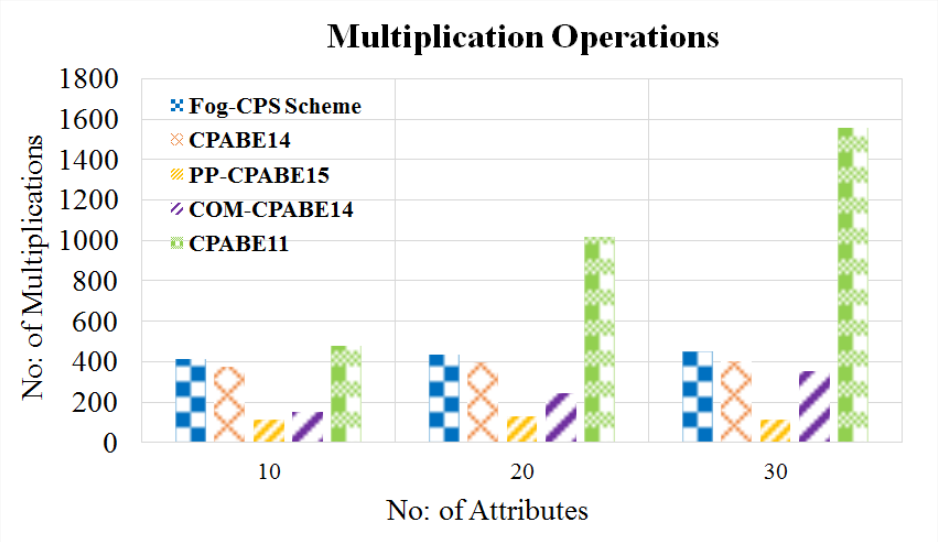


Fig. 4.20 No: of Multiplication Operations

schemes have been quantified. Again, ECC-Auth-18 requires the least number of operations i.e. 2 point additions and 5 scalar multiplications. The PP-CPABE15 and COM-CPABE14 schemes require 6 and 4 times fewer addition operations than the proposed scheme. The CPABE11 requires the highest number of additions and multiplications. Again, PP-CPABE15 and COM-CPABE14 require the least number of multiplications which is half as many as required by our scheme and CPABE14.

There are no divisions in the proposed scheme, whereas CPABE11 requires 5 times more division operations compared to PP-CPABE15 and COM-CPABE14. Lastly, CPABE11

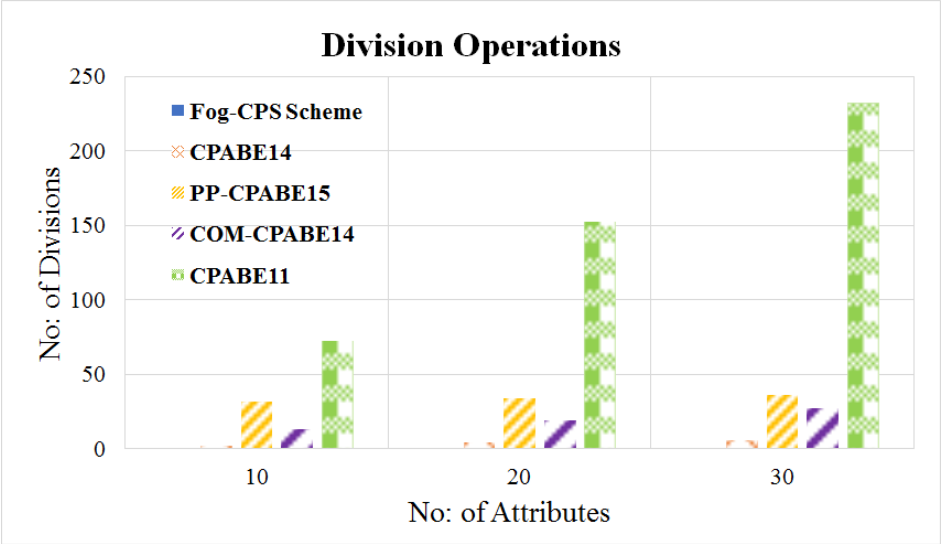


Fig. 4.21 No: of Division Operations

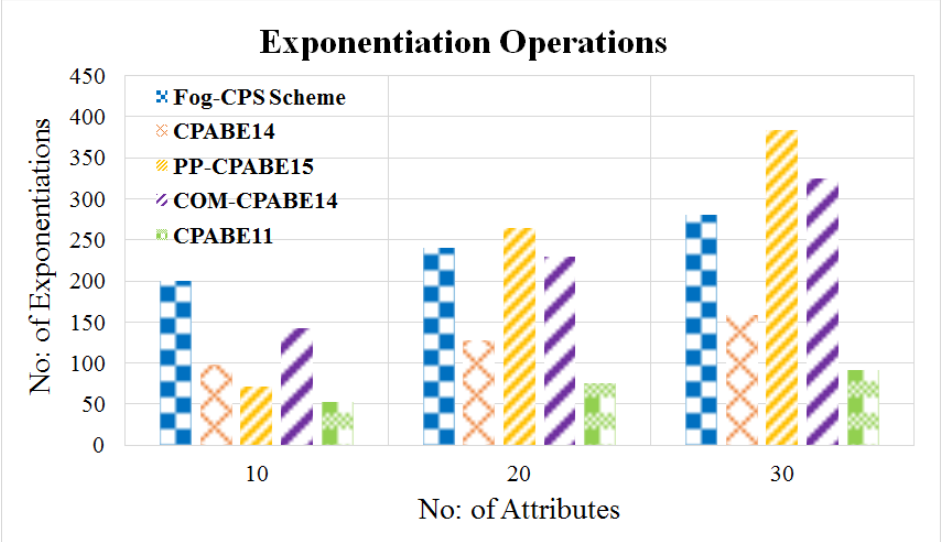


Fig. 4.22 No: of Exponentiation Operations

requires the least number of exponentiations which is 4 times less than this scheme. For exponentiation operation, the performance of the proposed scheme is comparable to PP-CPABE15 and COM-CPABE14. Compared to other ABE schemes, this scheme is computationally efficient because it requires no divisions and fewer addition and multiplication operations. ECC-Auth-18 is the most efficient scheme but it does not support access control. However, in the proposed scheme, the association of attributes and keys enables authentication and authorization and it therefore does not require any additional constructions.

Table 4.1 Memory Overhead

| Schemes | Setup | KeyGen | Partial KeyGen | Final KeyGen | Encrypt | Decrypt | KeyUpdate | Bytes |
|---|---|---|---|---|---|---|---|---|
| CPABE14 Guo et al. (2014) | $PK = (2n+1)\mathbb{G}_1 + \mathbb{G}_T$<br>$MSK = \mathbb{G}_1$ | $SK = 2\mathbb{G}_1$ | N/A | N/A | $(n-\lvert\mathbb{P}\rvert+2)\mathbb{G}_1 + \mathbb{G}_T$ | $M$ | N/A | 1636 |
| PP-CPABE15 Zhou et al. (2015) | $PK = (6n+1)\mathbb{G}_1$<br>$MSK = 2Z_p$ | $SK = (2\lvert\mathbb{A}\rvert+1)\mathbb{G}_1$ | N/A | N/A | $2\mathbb{G}_1 + \mathbb{G}_T$ | $M$ | N/A | 9876 |
| CPABE11 Chen et al. (2011) | $PK = 2n\mathbb{G}_1 + 2\mathbb{G}_T$<br>$MSK = 2n\mathbb{G}_1$ | $SK = (\lvert\mathbb{A}\rvert+1)\mathbb{G}_1$ | N/A | N/A | $2\mathbb{G}_1 + \mathbb{G}_T$ | $\mathbb{G}_T$ | N/A | 6932 |
| COM-CPABE14 Shota Yamada and Kunihiro (2014) | $PK = 6\mathbb{G}_1 + \mathbb{G}_T$<br>$MSK = 2Z_p$ | $SK = (4\lvert\mathbb{A}\rvert+2)\mathbb{G}_1$ | N/A | N/A | $3(\lvert\mathbb{P}\rvert+1)\mathbb{G}_1$ | $M$ | N/A | 6292 |
| ECC-CPABE Odelu and Das (2016) | $PK = (3n+1)\mathbb{G}$ | $SK = 2 \times O(P)$ | N/A | N/A | $(n^{\sim}\lvert P\rvert+3)\mathbb{G} + L$ | $M$ | N/A | 1760 |
| Fog-CPS Scheme | $N/A$ | N/A | $PK_{\mathbb{A}_S} = (n+1)\mathbb{G}$<br>$SK_{\mathbb{A}_S} = 1 \times O(P)$ | $PK_{\mathbb{A}_K} = (n+1)\mathbb{G}$<br>$SK_{\mathbb{A}_K} = 1 \times O(P)$ | $(n-\lvert\mathbb{P}\rvert+2)\mathbb{G}$ | $M$ | $2 \times O(P) + 2\mathbb{G}$ | 1340 |

Note: $\mathbb{G}_1$ and $\mathbb{G}_T$: prime order pairing groups (160-bit), *exp*: exponent operation, $\ddot{p}$: pairing operation, $\lvert\mathbb{A}\rvert$ and $\lvert\mathbb{P}\rvert$: No: of attributes in secret key and access policy, $\mathbb{G}$: elliptic curve group defined over finite field $Z_p$, $O(P)$: order of the base point (160-bit in $Z_p$), $L$: length of plaintext message $M$.

### 4.1.3 Memory Overhead

Table 4.1 shows the calculation of the memory overhead of each scheme. In MNT-159 curve, one group element in $\mathbb{G}$ and $\mathbb{G}_1$ take $2 \times 160 = 320$ bits whereas one group element in $\mathbb{G}_T$ takes $2 \times 512 = 1024$ bits. Likewise, in SS512 curve, one group element in $\mathbb{G}_1$ takes $2 \times 512 = 1024$ bits whereas one group element in $\mathbb{G}_T$ takes $2 \times 1024 = 2048$ bits.

The column **Bytes** represents the total number of bytes required in all algorithms (i.e., Setup, KeyGen, Encrypt, Decrypt, and KeyRevoke) for an attribute universe $\mathbb{U}$ of 10 and an access policy $\mathbb{P}$ of 5 attributes. In case of the proposed scheme, the memory overhead of partial/final key pair generation and update is also considered when quantifying the number of Bytes.

The proposed Fog-CPS scheme is lightweight than ECC-CPABE scheme on which it is based as it incurs less overhead. Our scheme requires $2(n+1)$ elements in $\mathbb{G}$ for generation of both partial and final public keys whereas ECC-CPABE requires $3(n+1)$ elements for public key. Likewise, for partial and final secret keys, our scheme requires only two secret elements in finite field $Z_p$ whereas ECC-CPABE requires three. The length of $CT$ in our scheme is $(n-|\mathbb{P}|+2)$ group $\mathbb{G}$ elements whereas $(n-|\mathbb{P}|+3)$ in ECC-CPABE.

As can be seen in Table 4.1, Fog-CPS scheme has the lowest memory overhead i.e. 1340 bytes followed by CPABE14 and ECC-CPABE scheme which take 1636 and 1760 bytes respectively. PP-CPABE15 Zhou et al. (2015) incurs the highest overhead of 9876 bytes. It is noted that for the proposed scheme, the memory overhead of one final public and secret key over a shared attribute set ($|\mathbb{A}_K| = 10$) has also been considered. Moreover, for *Setup* and *KeyGen* algorithms i.e. partial key pair generation in this case, Fog-CPS scheme has the lowest overhead compared to all other schemes. The final secret key in the proposed scheme only requires one element of the order of base point on the elliptic curve $\mathbb{G}$.

In CPABE14 Guo et al. (2014), there are $(2n+1)$ elements in $\mathbb{G}_1$ and one in $G_T$ for $PK$, two elements in $\mathbb{G}_1$ for $SK$, $(n-|\mathbb{P}|+2)$ elements in $\mathbb{G}_1$ for $CT$. In PP-CPABE15 Zhou et al. (2015), there are $(6n+1)$ $\mathbb{G}_1$ and $(2|\mathbb{A}|+1)$ group elements in $PK$ and $SK$ respectively, whereas $CT$ has 2 group elements in $\mathbb{G}_1$ and one in $\mathbb{G}_T$. The $PK$ in CPABE11 Chen et al. (2011) has $2n$ elements in $\mathbb{G}_1$ and two in $\mathbb{G}_T$ respectively, the $SK$ has $|\mathbb{A}|+1$ elements in $\mathbb{G}_1$, whereas the $CT$ has 2 elements in $\mathbb{G}_1$ and one in $\mathbb{G}_T$. In COM-CPABE14 Shota Yamada and Kunihiro (2014) scheme, $PK$ contains 6 elements in $\mathbb{G}_1$ and one in $G_T$, $SK$ contains $(4|\mathbb{A}|+2)$ elements in $\mathbb{G}_1$, and $CT$ contains $3(|\mathbb{P}|+1)$ in $\mathbb{G}_1$. The PP-CPABE15 Zhou et al. (2015) has the highest memory overhead followed by CPABE11 Chen et al. (2011) and COM-CPABE14 Shota Yamada and Kunihiro (2014). Both PP-CPABE15 Zhou et al. (2015) and CPABE11 Chen et al. (2011) have constant size ciphertexts which only require 2 group elements in $\mathbb{G}_1$ and one element in $\mathbb{G}_T$. The $CT$ in Fog-CPS scheme and CPABE14 Guo

et al. (2014) have almost the same number of elements. In key update algorithms 6, 7 and 8, this scheme only requires 2 elements in $\mathbb{G}$ and 2 elements of the order of base point on the elliptic curve $\mathbb{G}$.

Table 4.2 Computational Overhead

| Schemes | Setup | KeyGen | Final KeyGen | Encrypt | Decrypt | KeyRevoke | Total Operations |
|---|---|---|---|---|---|---|---|
| CPABE14 Guo et al. (2014) | $(2n)exp+\ddot{p}$ | $2exp$ | N/A | $(n-\lvert\mathbb{P}\rvert+3)exp+\ddot{p}$ | $(2\lvert n-\mathbb{P}\rvert+4)exp+4\ddot{p}$ | N/A | 44 $exp$ + 6 $\ddot{p}$ |
| PP-CPABE15 Zhou et al. (2015) | $(6n+1)exp$ | $(2\lvert\mathbb{A}\rvert+1)exp$ | N/A | $3exp+\ddot{p}$ | $(2\lvert\mathbb{P}\rvert+1)exp+(2\lvert\mathbb{P}\rvert+1)\ddot{p}$ | N/A | 96 $exp$ + 12 $\ddot{p}$ |
| CPABE11 Chen et al. (2011) | $nexp+n\ddot{p}$ | $\lvert\mathbb{A}\rvert exp$ | N/A | $3exp$ | $2\ddot{p}$ | N/A | 23 $exp$ + 12 $\ddot{p}$ |
| COM-CPABE14 Shota Yamada and Kunihiro (2014) | $1exp$ | $(4\lvert\mathbb{A}\rvert+2)exp$ | N/A | $(3\lvert\mathbb{P}\rvert+1)exp$ | $2\lvert\mathbb{P}\rvert exp+2\lvert\mathbb{P}\rvert\ddot{p}$ | N/A | 69 $exp$ + 10 $\ddot{p}$ |
| ECC-CPABE Odelu and Das (2016) | $3(n+1)ecm$ | N/A | N/A | $(n-\mathbb{P}+2)ecm$ | $(n-\mathbb{P}+3)ecm$ | N/A | 48 $ecm$ |
| Fog-CPS Scheme | $N/A$ | $(n+1)ecm$ | $(n+1)ecm$ | $(n-\mathbb{P}+1)ecm$ | $(n-\mathbb{P}+2)ecm$ | 2 $ecm$ | 37 $ecm$ |

Note: $exp$: exponentiation operation, $\ddot{p}$: pairing operation, and $ecm$: scalar point multiplication in the elliptic curve group $\mathbb{G}$

### 4.1.4 Computational Overhead

The computational overhead of all schemes is summarized in Table 4.2. The column **Total Operations** represent the total number of operations by considering all algorithms i.e. Setup, KeyGen, Encrypt and Decrypt. In the proposed scheme, the computational overhead of the partial and final key pair generation and update are also considered. From table 4.2, it is observed that Fog-CPS scheme introduces lowest computational overhead than all other schemes which are based on bilinear maps and elliptic curves.

For partial and final public key generation, Fog-CPS scheme requires $(2n+2)$ scalar multiplications in the elliptic curve group $\mathbb{G}$. Likewise, for *Encrypt* and *Decrypt*, $(n-\mathbb{P}+1)$ and $(n-\mathbb{P}+2)$ scalar point multiplications are required respectively. Moreover, in key update algorithms i.e. 6, 7, and 8, this scheme only requires two scalar multiplications.

For the *Setup* and *KeyGen* algorithms, CPABE14 Guo et al. (2014) requires $(2n+2)$ exponentiations and 1 pairing operation respectively. In addition, *Encrypt* algorithm requires $(n-|\mathbb{P}|+3)$ exponentiations and a single pairing whereas *Decrypt* requires $(2|n-\mathbb{P}|+4)$ exponentiations and 4 pairing operations. The *Setup* and *KeyGen* algorithms in PP-CPABE15 Zhou et al. (2015) require $6n+1$ and $(2|\mathbb{A}|+1)$exponentiation operations respectively. Similarly, for *Encrypt*, PP-CPABE15 requires 3 exponentiations and 1 pairing, whereas for *Decrypt* $(2|\mathbb{P}|+1)$ exponentiation and pairing operations are required. Moreover, for *Setup* and *KeyGen* algorithms, CPABE11 Chen et al. (2011) requires $(n+|\mathbb{A}|)$ exponentiation and $n$ pairing operations. On the contrary, CPABE11 requires, 3 exponentiation and 2 pairing operation for *Encrypt* and *Decrypt* algorithms respectively. In COM-CPABE14 Shota Yamada and Kunihiro (2014), $(4|\mathbb{A}|+3)$ exponentiations are required for both *Setup* and *KeyGen* algorithms. However, *Encrypt* algorithm requires $(3|\mathbb{P}|+1)$ exponentiations and *Decrypt* requires $2|\mathbb{P}|$ exponentiation and pairing operations.

Table 4.3 Simulation Parameters

| **Parameters** | **Values** |
|---|---|
| No: of Fog Nodes | 2 |
| No: of CPS devices connected with each fog node | 20 |
| No: of FA nodes | 1 |
| No: of regions | 2 |
| No: of Cloud Providers | 1 |
| No: of Simulation Executions | 30 |

## 4.2 Experimental Evaluation of TMS

This experiment is designed to achieve three objectives:

1. computing trust for Fog-CPS entities based on the proposed TMS
2. demonstrating the effectiveness of the credibility model to compute precise and accurate trust
3. integration of SC and TMS components

### 4.2.1 Implementation Environment

The experimental setup is same as the one discussed in section 4.1.1 meaning that the TMS is executed both on a raspberry Pi and a desktop computer. For running TMS on raspberry Pi, Octave (GNU, 2019) and a few machine learning libraries are installed. Moreover, for desktop computer experiments, the random forest regression model is trained and tested in Spyder 3.2.6, it is a scientific Python development environment which is packaged in Anaconda. Trust results are same in both experimental setups but the processing time is different. The experiments on the raspberry Pi are slower than the ones on desktop computer.

### 4.2.2 Dataset Generation

As discussed in section 3.6, the Fog-CPS system consists of three layers, *CPS devices*, *fog nodes* and *cloud*. Communication between these layers is possible in four different ways:

1. device to device
2. device to fog node
3. fog node to fog node
4. fog node to cloud service provider

For evaluating the proposed TMS, a generalized Fog-CPS network was simulated (see Figure 4.1) in iFogSim (Gupta et al., 2017). The simulation parameters are listed in Table 4.3. It can be observed in Figure 4.1 that there is one cloud service provider and one FA. Moreover, there is one fog node in each region of *fog* layer. Twenty CPS devices are provisioning services from each of the fog nodes. CPS devices belong to different Fog- CPS application scenarios, namely weather forecasting, health monitoring, energy consumption and vehicular ad hoc networks (VANETs) etc. For experimental purposes, the simulation model quantifies the multidimensional QoS parameters in the following three cases:

1. CPS device to fog node communication
2. fog node to CPS device communication
3. FA monitoring fog nodes

A communication loop is created wherein a CPS device provisions a service (i.e. compute, storage, network, and software) from a fog node and subsequently reports its parameters. As discussed in section 3.6.4, the trust of fog nodes is computed by aggregating $T^{fa \to fog}$ and $T^{cps \to fog}$. So, for computing objective and CPS device trust, the parameters listed in Table 3.2 are measured. It is noted that the parameter measurement takes place at discrete time instances, after the fog node finishes one of the assigned task. Both FA and the CPS device which is provisioning service from that fog node record their respective set of parameters. FA quantifies the average CPU (GHz percentage), memory (GB), disk (TB) utilization, average task success ratio, and average response time (ms). The "CPS Device Parameter Monitoring" module installed in the CPS device quantifies the response time (sec), bandwidth (bit/sec), and energy consumption (Joules) when communicating to the fog node.

The simulated Fog-CPS system is executed 30 times to quantify multidimensional features. As 20 CPS devices are connected to each fog node, in each simulation, every device is sending 20 reports thus totalling to 400 reports for one fog node. Eventually, for both fog nodes, 24,000 reports are sent in 30 executions of simulation. Likewise, FA also monitors the service quality once during each run of simulation thus generating 60 samples for both fog nodes. The parameters are further used as features in random forest regression model.

### 4.2.3 Trust Label Generation

Having generated the dataset, the next task is to generate the trust labels corresponding to the set of parameters. The iFogSim (Gupta et al., 2017) simulator quantifies the QoS parameters but it does not generate their corresponding trust labels. In order to do so, all the acquired QoS parameters are averaged to find the normality and based on which the trust labels (i.e. trust degree $T_t$ at time instance $t$) are predicted by employing random forest regression for each set of the service features. Precisely, the instant trust degree of objective trust $T^{fa \to fog}$ gets a high value if the average task success ratio is high and the fog node fulfilled the service request by maintaining the acceptable service quality and vice versa. Likewise, the CPS Device trust $T^{cps \to fog}$ assigned a higher value if the parameters reported by CPS devices are in a given range which is quantified by averaging the parameter values in 30 runs of simulations.

### 4.2.4 Random Forest Regression Training and Testing

As discussed in Section 3.6.7, that random forest regression is employed to compute the trust of fog nodes and CPS devices. Next, discussion moves to how the regression model is trained and tested for accurate prediction of trust. Random Forest Regression model from "sklearn. ensemble" with parameters n_estimators = 50 and max_depth = 6 is used.

#### Train – Test Split

Next, the dataset is partitioned and 70% of the samples are used for training whereas 30% are used for testing. After intensive training of the regression model, it is tested with the remaining 30% of the data samples. In the proposed TMS, Random Forest Regression is employed to predict both objective trust $T^{fa \rightarrow fog}$ and CPS device trust $T^{cps \rightarrow fog}$. The mean square error (MSE) is zero (0) in case of $T^{fa \rightarrow fog}$ as this dataset is very small having only 60 samples, out of which 70% (42 samples) are used for training and rest (18) for testing. So, achieving zero MSE with a non-linear regression model is justifiable. However, it was also expected that the regression model might show different accuracy results with a bigger dataset. In the case of CPS device trust $T^{cps \rightarrow fog}$, for example, the MSE is 0.12 meaning that the model predicted 88 % of the trust labels accurately.

### 4.2.5 Trust Results

Having trained the regression model, trust of fog nodes and CPS devices was computed based on the equations 3.27, 3.29 and 3.32 discussed in Section 3.6. Additionally, the integration of two components is elaborated by embedding trust of an entity in an access policy and subsequently describing its corresponding access control rights. The experimental results are divided into following categories:

1. Fog-CPS Entities Trust Results
2. Credibility Model Evaluation
3. Integration of SC and TMS
4. Comparative Analysis

### 4.2.6 Fog-CPS Entities Trust Results

In this experiment, it is assumed that both fog nodes and CPS devices operate legitimately conforming the system/protocol specifications of Fog-CPS. The QoS parameters are quanti-

Table 4.4 Trust Results

| ID | $T^{cps \to fog}$ | $T^{fa \to fog}$ | $T^{fog}$ | Time |
|---|---|---|---|---|
| $fog_1$ | 0.60 | 0.59 | 0.70 | $t_1$ |
| | 0.64 | 0.67 | 0.74 | $t_2$ |
| | 0.80 | 0.57 | 0.89 | $t_3$ |
| | 0.69 | 0.77 | 0.74 | $t_4$ |
| | 0.59 | 0.60 | 0.73 | $t_5$ |
| | 0.54 | 0.63 | 0.69 | $t_6$ |
| $fog_2$ | 0.60 | 0.50 | 0.79 | $t_1$ |
| | 0.58 | 0.90 | 0.58 | $t_2$ |
| | 0.80 | 0.55 | 0.67 | $t_3$ |
| | 0.53 | 0.58 | 0.63 | $t_4$ |
| | 0.75 | 0.78 | 0.82 | $t5$ |
| | 0.59 | 0.76 | 0.83 | $t_6$ |

fied in six different time periods with an increment of 10 minutes in each subsequent time instance. Precisely, the first time period was 20 minutes, the second 30 minutes, and so forth. The trust values of both fog nodes and CPS devices are presented. For fog nodes, there are three results, namely CPS trust $T^{cps \to fog}$, objective trust $T^{fa \to fog}$ and fog node trust $T^{fog}$ listed in Table 4.4. Hereinafter, the notations $cps_i$, $fog_i$, $t_i$ denote ith CPS device, fog node, and time instance respectively.

### 1. CPS Device Trust $T^{cps \to fog}$

Figure 4.23 illustrates the trust of a CPS device for fog node $fog_1$ based on energy consumption, response time and bandwidth predicted using the random forest regression model. The trust of a CPS device in first time instance $t_1$ is 0.79, $t_2$ is 0.84, $t_3$ is 0.72, $t_4$ is 0.78, $t_5$ is 0.62, and $t_6$ is 0.74.

Figure 4.23 also illustrates the final CPS trust $T^{cps \to fog}$ for fog nodes $fog_1$ and $fog_2$ computed using Eq. (3.29) presented in Section 3.6.11. The final CPS trust $T^{cps \to fog}$ of fog node $fog_1$ in time instance $t_1$ is 0.60, $t_2$ is 0.64, $t_3$ is 0.80, $t_4$ is 0.69, $t_5$ is 0.59, and $t_6$ is 0.54. Similarly, the trust scores of fog node $fog_2$ at six time instances are 0.60, 0.58, 0.80, 0.53, 0.75, and 0.59. As can be seen from Figure 4.23, in all time instances, the CPS trust $T^{cps \to fog}$ of both fog nodes $fog_1$ and $fog_2$ is trustworthy i.e. greater than threshold 0.5.

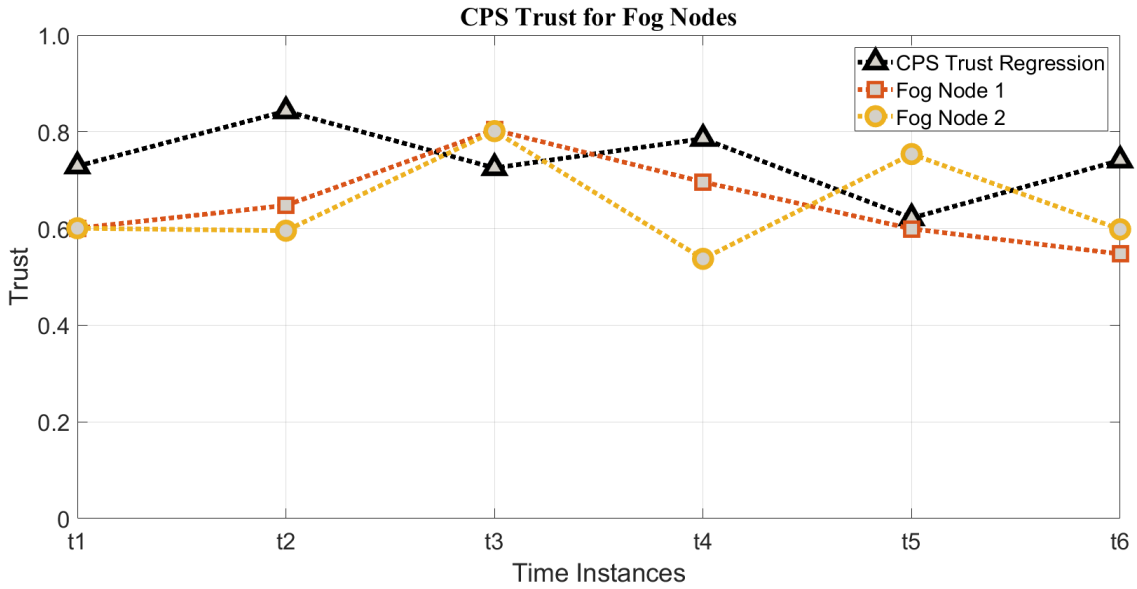


Fig. 4.23 CPS Trust for Fog Nodes in Hostile Free Environment

### 2. Objective Trust $T^{fa \rightarrow fog}$

Figure 4.24 lists the objective trust $T^{fa \rightarrow fog}$ values of fog nodes computed using Eq. (3.27) presented in Section 3.6.7. The objective trust $T^{fa \rightarrow fog}$ scores of first fog node $fog_1$ in discrete time instances $[t_1 - t_6]$ are 0.59, 0.67, 0.57, 0.77, 0.60, and 0.63 whereas the trust scores of second fog node $fog_2$ are 0.50, 0.90, 0.55, 0.58, 0.78, and 0.76. It can be seen that in the 1st time instance $t_1$, the fog node $fog_1$ has slightly higher objective trust $T^{fa \rightarrow fog}$ than fog node $fog_2$. In the 2nd time instance $t_2$, $fog_2$ has higher objective trust $T^{fa \rightarrow fog}$ than $fog_1$ whereas in the 3rd time instance $t_3$, the objective trust $T^{fa \rightarrow fog}$ of both fog nodes is almost equal. Moreover, in the 4th time instance the objective trust $T^{fa \rightarrow fog}$ of $fog_1$ is again greater than fog node $fog_2$. Likewise in the 5th and 6th time instances, the objective trust $T^{fa \rightarrow fog}$ of fog node $fog_2$ is greater than fog node $fog_1$. Overall as per the QoS evidence, the performance of both fog nodes is trustworthy.

### 3. Fog Node Trust $T^{fog}$

Having computed the CPS and objective trust scores, the FA aggregates them to compute the final trust of fog nodes $T^{fog}$ using Eq. (3.32) presented in Section 3.6.13. Again according to the assumption, both the fog nodes and CPS devices operate honestly and therefore assigned equal weight $\delta = 0.5$ in Eq. (3.32). Figure 4.25 illustrates the final trust $T^{fog}$ of two fog nodes. The $T^{fog}$ of fog node $fog_1$ in six time instances $[t_1, t_2, \ldots, t_6]$ is 0.70, 0.74, 0.89, 0.74,

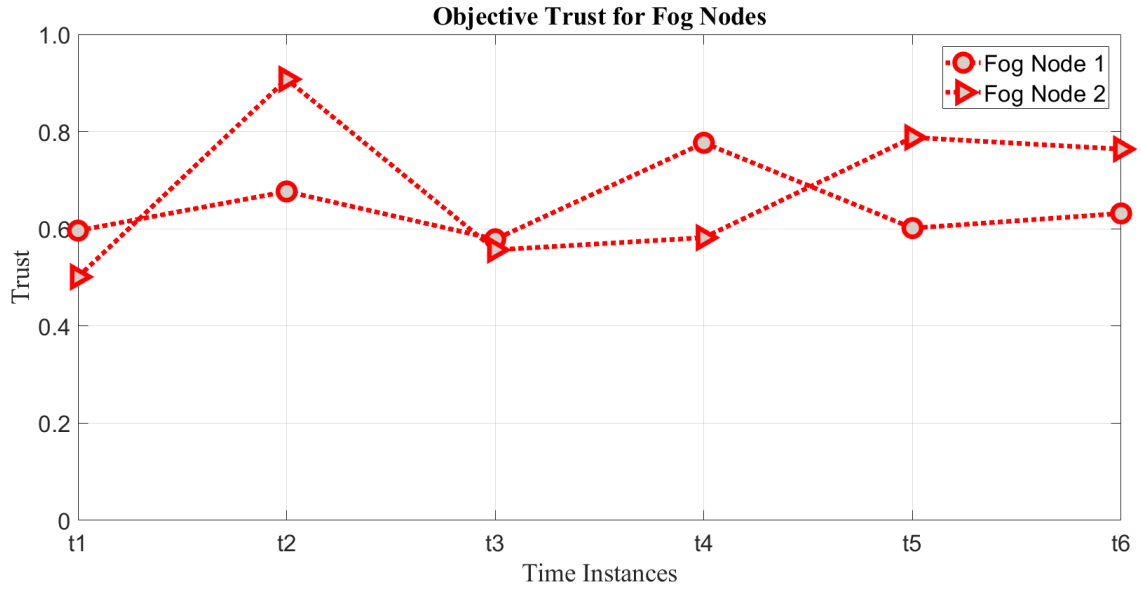


Fig. 4.24 Objective Trust for Fog Nodes in Hostile free Environment

0.73, and 0.69 respectively. Similarly, the $T^{fog}$ of fog node $fog_2$ in all six time instances is 0.79, 0.58, 0.67, 0.63, 0.82, and 0.83.

### 4. Trust Computation for CPS Devices

FA also computes a trust score of all CPS devices which are getting services from different fog nodes. Both fog nodes and CPS devices assess each other on the same set of parameters i.e. energy consumption, bandwidth and response time.

Table 4.5 Access Control Rights based on Fog Node Trust

| IDs | $T^{fog}$ at Different Time Instances | | | | | | Access Control Rights |
|---|---|---|---|---|---|---|---|
| | $t_1$ | $t_2$ | $t_3$ | $t_4$ | $t_5$ | $t_6$ | |
| $fog_1$ | 0.7 | 0.74 | 0.89 | 0.74 | 0.73 | 0.69 | Special Permission |
| $fog_2$ | 0.79 | 0.58 | 0.67 | 0.63 | 0.82 | 0.83 | Read and Execute |

### 5. Trust and Access Control Rights

Table 4.5 lists the $T^{fog}$ of fog nodes in five time instances. Trust determines the access control rights granted to each fog node. For brevity of expression, the access right in the last time instance is listed. In normal circumstances, the trust would not change very frequently and therefore the access rights. However, monitoring the rate of change of trust can assist in detection and prevention of malicious activities.

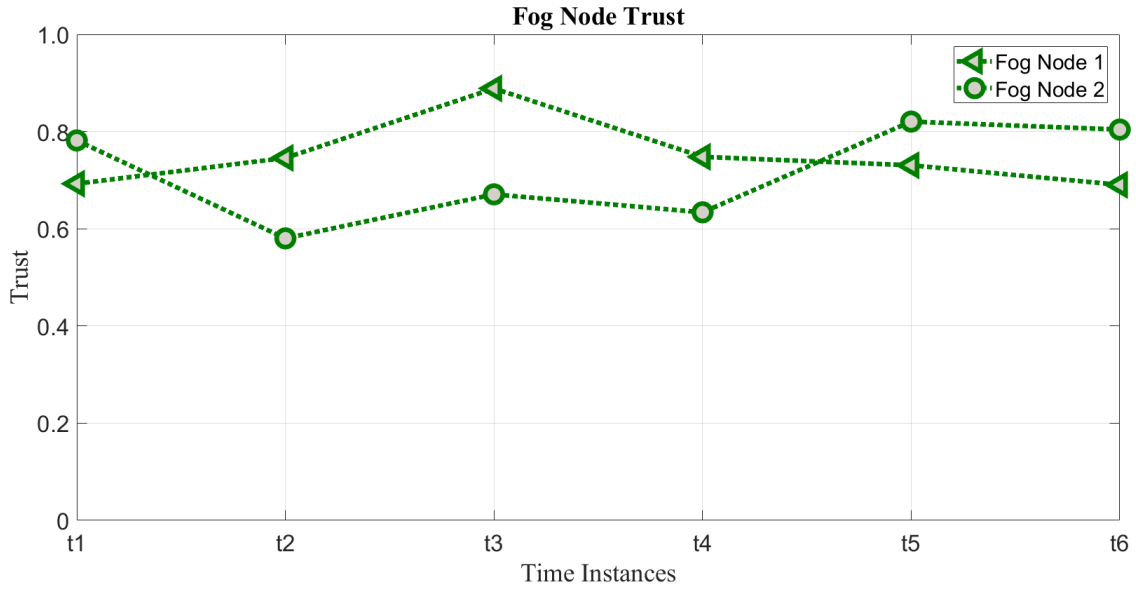


Fig. 4.25 Fog nodes Trust in Hostile Free Environment

Table 4.6 Collusion Attack Scenarios

| Attack | Description |
|---|---|
| $A_1$ | No Malicious CPS Device Parameters |
| $A_2$ | 10% Malicious CPS Device Parameters |
| $A_3$ | 25% Malicious CPS Device Parameters |
| $A_4$ | 50% Malicious CPS Device Parameters |
| $A_5$ | 75% Malicious CPS Device Parameters |
| $A_6$ | 100% Malicious CPS Device Parameters |

### 4.2.7 Credibility Model Evaluation

In section 3.6.12, the compromise of Fog-CPS entities and its impact on trust computations was discussed. This experiment is designed to evaluate the effectiveness of the credibility model in maintaining the accuracy of trust computation model even in presence of malicious/compromised entities. The credibility in **Case-1** is evaluated wherein the the CPS devices are considered compromised. However, similar results will be produced for compromised FA and Fog nodes when the environment changes. A hostile environment is setup by considering six scenarios for collusion attacks wherein the trust of fog nodes is computed. In each attacking scenario, a percentage of parameter reports sent by CPS devices are considered malicious. Hereinafter, the notation $A_i$ is used to denote $i$th attacking scenario. Table 4.6 lists the different collusion attacking scenarios.

In the ***first*** attacking scenario $A_1$, there are 0% malicious parameters i.e. all CPS devices are honest. In the ***second*** attacking scenario $A_2$, 10% parameters are considered malicious. Likewise, in the ***third*** $A_3$, ***fourth*** $A_4$, and ***fifth*** $A_5$ attacking scenarios, 25%, 50%, and 75% of parameters are malicious. Lastly, in the ***sixth*** attack scenario $A_6$, all parameters are malicious. The attacking scenarios are designed such that the effectiveness of credibility model can be evaluated in different configurations of hostile environments. Subsequently, it is demonstrated how the proposed trust computation model maintains the accuracy of trust results even in presence of malicious CPS devices.

The trust credibility model is evaluated in two cases where CPS devices send parameters with high and low values to change the trust of fog nodes. Two cases, $C_1$ and $C_2$, of credibility evaluation are formulated as follows:

1. In the ***first*** case, $C_1$, the malicious CPS devices send parameters with very high values in the range of [0.8 - 1] and very low values in the range of [0.05 - 0.2] in all attacking scenarios.

2. In the ***second*** case, $C_2$, the CPS devices send parameters with an increment and decrement of 0.1 (i.e slightly changing the values from threshold value of 0.5). Precisely in each attacking scenario, the respective percentage of malicious CPS devices send parameters reports with an increment and decrements of 0.1.

The credibility model analyzes the rate of change in $T^{cps \rightarrow fog}$ in the consecutive time durations $[\tau-1, \tau]$ and subsequently adjusts the CPS trust in current time duration using Eq. (3.30). However, for finding an appropriate value of $\acute{\sigma}$, the standard deviation in a hostile free environment during a time duration $\tau$ consisting of six time instances $[t_1, t_2, \ldots, t_6]$ is computed using Eq. (3.31). Similarly, the standard deviation in two hostile environments (i.e. credibility cases) is also computed. Lastly, final standard deviation is computed by the difference among three standard deviations. Following above computational procedure, $\acute{\sigma} = 0.03$ was identified and subsequently used in the credibility evaluation model. Having discussed the hostile environment, the impact of credibility model in accurate and precise computation of $T^{cps \rightarrow fog}$ and $T^{fog}$ is explained. For clarity, $T^{cps \rightarrow fog}$ and $T^{fa \rightarrow fog}$ results of only first fog node $fog_1$, with or without considering the credibility model are presented. Additionally, the change in access control rights of $fog_1$ based on the change in trust in each attacking scenario in the first credibility case $C_1$ is discussed in Tables 4.7 and 4.8.

### 1. Credibility Evaluation Case-1

The ***first*** case, $C_1$, elaborates how the credibility model maintains accurate and precise computation of CPS trust $T^{cps \rightarrow fog}$ and fog node trust $T^{fog}$. Both results with or without

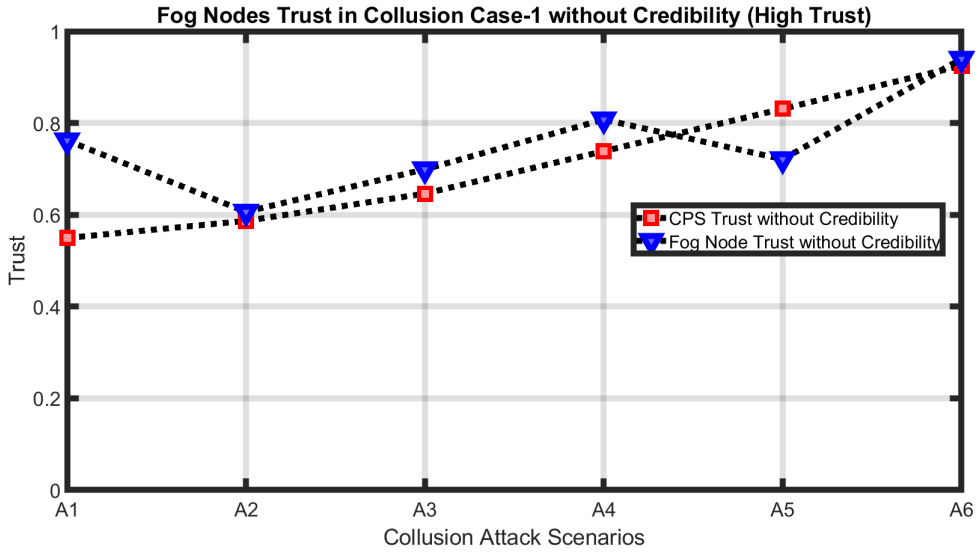


Fig. 4.26 Fog Nodes High Trust without Credibility in Hostile Environment (***First*** Case $C_1$)

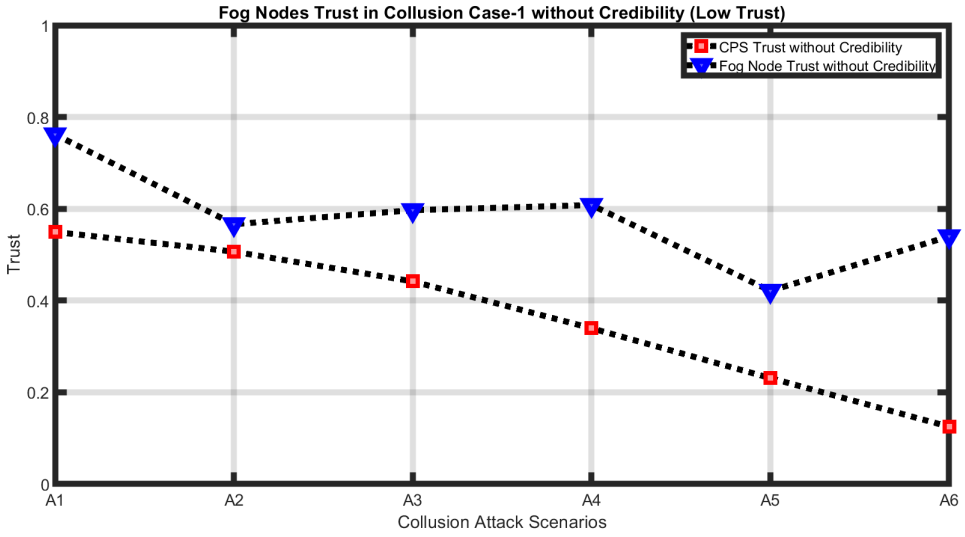


Fig. 4.27 Fog Nodes Low Trust without Credibility in Hostile Environment (***First*** Case $C_1$)

credibility are presented. It is noted that the attacking scenarios take place in different time instances i.e. $[t_1, t_2, \ldots, t_6]$. However, the results do not mention the time instances but they should be considered when analyzing the results.

### Without Credibility

$T^{cps \rightarrow fog}$ and $T^{fog}$ trust results without considering the credibility are shown in Figures 4.26 and 4.27. When malicious CPS devices report high trust values, $T^{cps \rightarrow fog}$ increases in each

subsequent attacking scenario as shown in Figure 4.26. As a result, $T^{fog}$ is also increasing. Table 4.7 lists the change in access control rights based on change in $T^{cps \rightarrow fog}$ and $T^{fog}$. It can be observed that access control rights of $fog_1$ escalate from "Modify" to "All" in case of high trust, when trust is computed without considering the credibility model.

Similarly, in case of low trust values as shown in Figure 4.27, the $T^{cps \rightarrow fog}$ and $T^{fog}$ are decreasing with increasing percentage of malicious CPS devices in different attacking scenarios. Likewise, in case of low trust, the change in access control rights based on change in $T^{cps \rightarrow fog}$ and $T^{fog}$ from "Modify" to "Delete" can be observed in Table 4.8. The consequences of such a dramatic change in access control rights can lead to catastrophic results in a real Fog-CPS system.

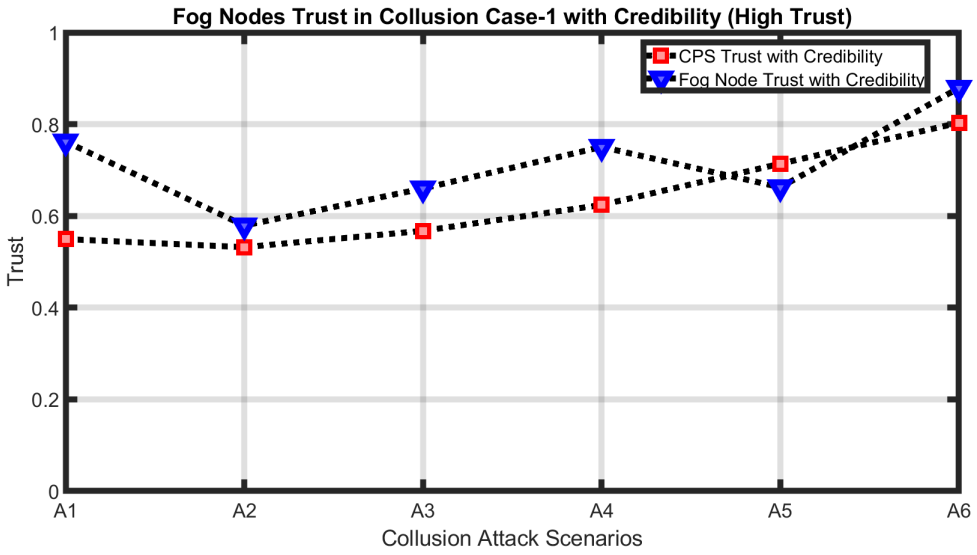


Fig. 4.28 Fog Nodes High Trust with Credibility in Hostile Environment (***First*** Case $C_1$)

### With Credibility

Figures 4.28 and 4.29 illustrate the CPS trust $T^{cps \rightarrow fog}$ and fog node trust $T^{fog}$ computed by considering the credibility model in the first case. From Figure 4.28, it can be observed that CPS trust is increasing with high trust values in each attacking scenario due to increasing percentage of malicious devices. It has increased from 0.54 in $A_1$ to 0.80 in $A_6$ which subsequently increased the fog node trust $T^{fog}$. However, due to the credibility model, there has not been a dramatic increase in trust. Same goes with the access control rights, Table 4.7 lists the change in access control rights in case of high trust. In contrast to the without credibility results, this time the privilege escalation is rather slow,i.e. from "Modify"

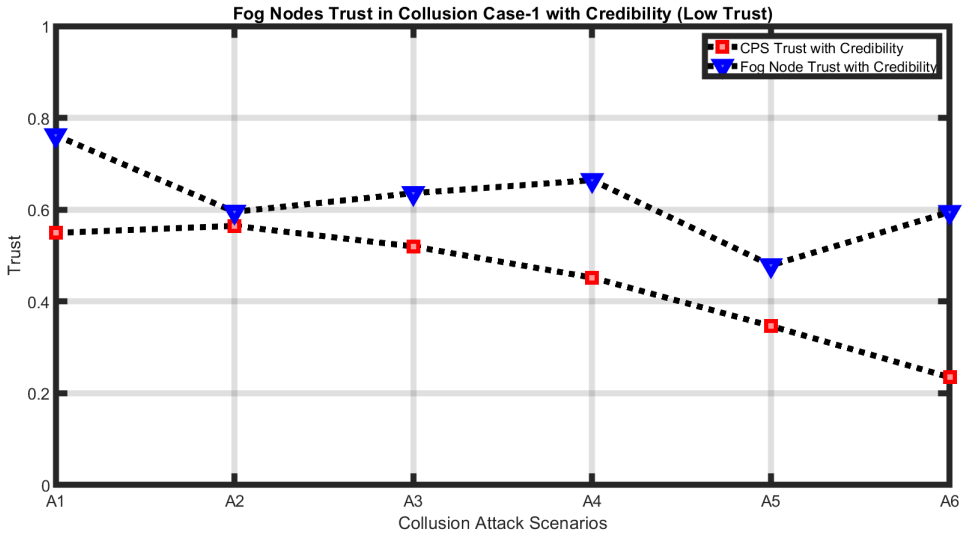


Fig. 4.29 Fog Nodes Low Trust with Credibility in Hostile Environment (***First*** Case $C_1$)

to "Special". But even this level of privilege escalation can cause havoc in a large-scale Fog-CPS system.

Likewise, Figure 4.29 shows the CPS trust $T^{cps \to fog}$ and fog node trust $T^{fog}$ computed when the malicious CPS devices send low values of trust. Again, it can be observed that CPS trust $T^{cps \to fog}$ sharply decreases from 0.54 in $A_1$ to 0.23 in $A_6$. Fog node trust $T^{fog}$ is also changing, due to change in $T^{cps \to fog}$, it dropped from 0.73 to 0.60. In case of low trust, trust results computed based on the proposed credibility model and the corresponding access rights of $fog_1$ are listed in Table 4.8.

It can be analyzed that without the credibility model malicious CPS devices can easily increase/decrease the trust of fog nodes and subsequently push trust to highest 1 and lowest 0 values. It is therefore essential to compute the credibility of $T^{cps \to fog}$ and adjust any discrepancies accordingly.

Table 4.7 Trust Credibility and Access Rights (High Trust)

| **Attacking Scenarios** | **Without Credibility** | | | **With Credibility** | | |
|---|---|---|---|---|---|---|
| | $T^{cps \to fog}$ | $T^{fa \to fog}$ | **ACR** | $T^{cps \to fog}$ | $T^{fa \to fog}$ | **ACR** |
| $A_1$ | 0.54 | 0.76 | Modify | 0.54 | 0.76 | Modify |
| $A_2$ | 0.58 | 0.60 | Delete | 0.53 | 0.57 | Delete |
| $A_3$ | 0.64 | 0.69 | Read and Execute | 0.56 | 0.69 | Read and Execute |
| $A_4$ | 0.73 | 0.8 | Modify | 0.62 | 0.76 | Modify |
| $A_5$ | 0.83 | 0.72 | Modify | 0.71 | 0.66 | Read and Execute |
| $A_6$ | 0.94 | 0.94 | All | 0.8 | 0.87 | Special Permission |

Note: ACR - Access Control Rights

Table 4.8 Trust Credibility and Access Rights (Low Trust)

| **Attacking Scenarios** | **Without Credibility** | | | **With Credibility** | | |
|---|---|---|---|---|---|---|
| | $T^{cps \to fog}$ | $T^{fa \to fog}$ | **ACR** | $T^{cps \to fog}$ | $T^{fa \to fog}$ | **ACR** |
| $A_1$ | 0.54 | 0.76 | Modify | 0.54 | 0.76 | Modify |
| $A_2$ | 0.50 | 0.56 | Delete | 0.56 | 0.59 | Delete |
| $A_3$ | 0.44 | 0.59 | Delete | 0.51 | 0.63 | Read and Execute |
| $A_4$ | 0.33 | 0.6 | Delete | 0.45 | 0.66 | Read and Execute |
| $A_5$ | 0.23 | 0.42 | Write | 0.34 | 0.47 | Write |
| $A_6$ | 0.12 | 0.54 | Delete | 0.23 | 0.59 | Delete |

Note: ACR - Access Control Rights

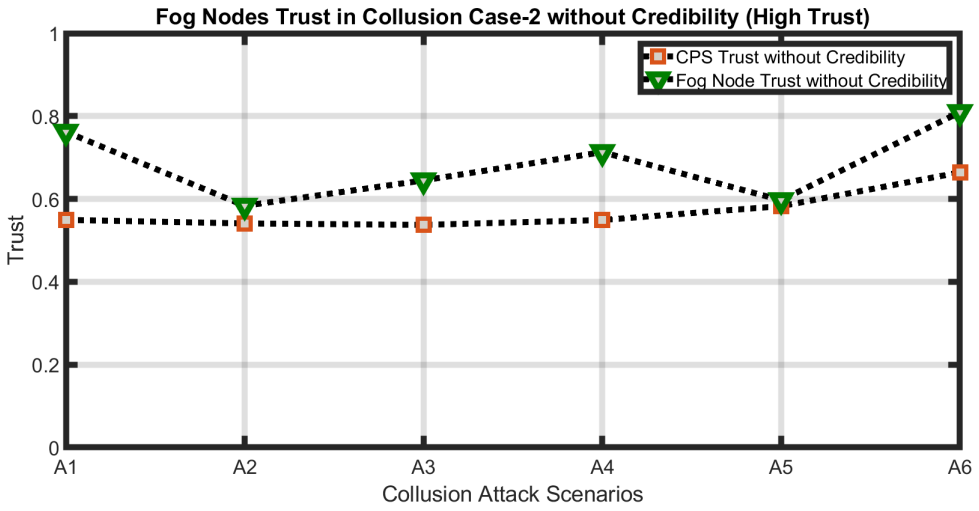


Fig. 4.30 Fog Nodes High Trust without Credibility in Hostile Environment (***Second*** Case $C_2$)

### 2. Credibility Evaluation Case-2

The ***second*** case $C_2$ of the credibility evaluation is designed to check the robustness of credibility model in detecting smaller changes in CPS trust. In this experiment, the malicious devices are slightly changing the parameters values with an increment and decrement of 0.1. In other words, in case of high trust, if in $A_1$, all devices are sending parameter values between 0.5 and 0.6. In the second attacking scenario $A_2$, 10% would send values between 0.6 and 0.7 while the rest of them will report values between 0.5 and 0.6. Likewise, in $A_3$, 25% would send values between 0.7 and 0.8 while the rest of them will report values between 0.5 and 0.6. The same happens to low trust wherein CPS devices try to slightly decrease the trust in each subsequent attacking scenario.

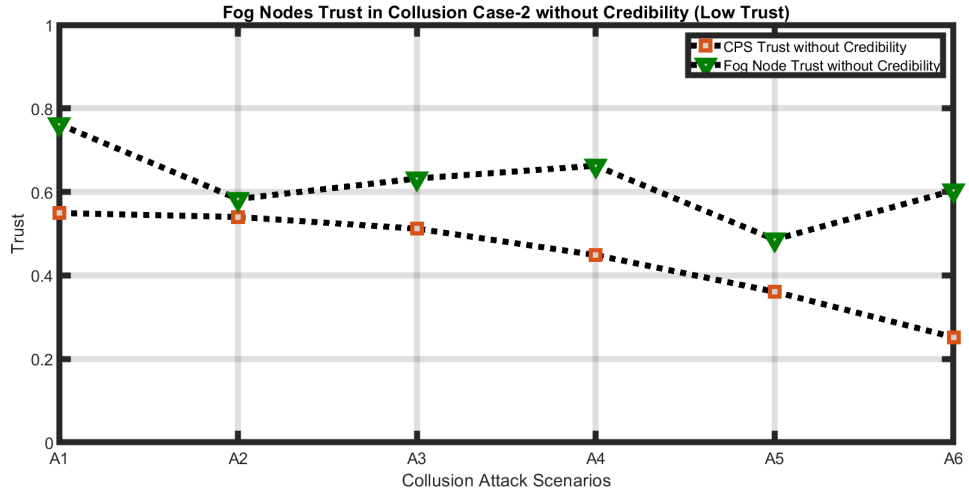


Fig. 4.31 Fog Nodes Low Trust without Credibility in Hostile Environment (***Second*** Case $C_2$)

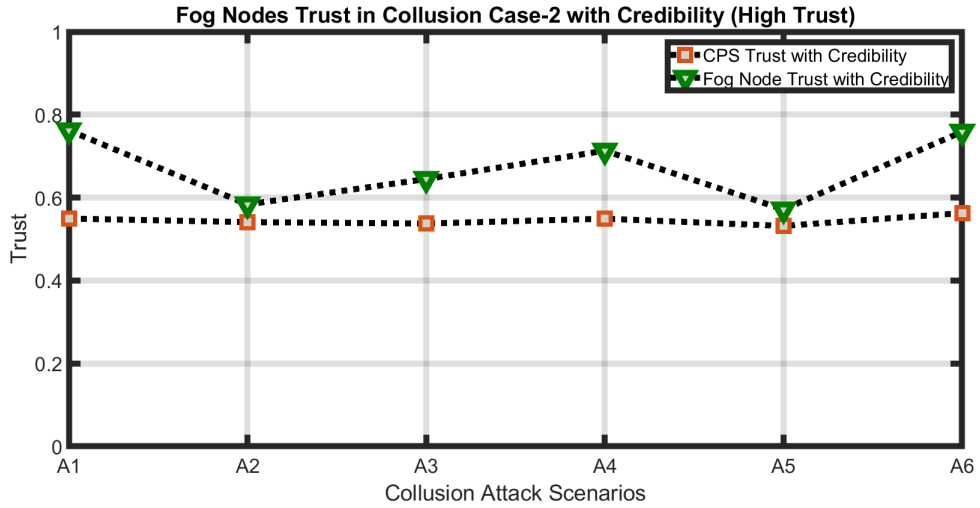


Fig. 4.32 Fog Nodes High Trust with Credibility in Hostile Environment (***Second*** Case $C_2$)

### Without Considering the Credibility Model

Figure 4.30 shows the CPS and fog node high trust computed without considering the credibility model. It can be analyzed that there has been a slight increase in CPS trust $T^{cps \rightarrow fog}$ in each attacking scenario. CPS trust increases from 0.54 in $A_1$ to 0.66 in $A_6$. Fog node trust $T^{fog}$ is also changing accordingly. Similar to high trust, the malicious CPS devices can also collude to decrease the $T^{cps \rightarrow fog}$. Figure 4.31 shows the trust when CPS devices are sending parameter values which result into lower trust. Again, it can be observed that CPS

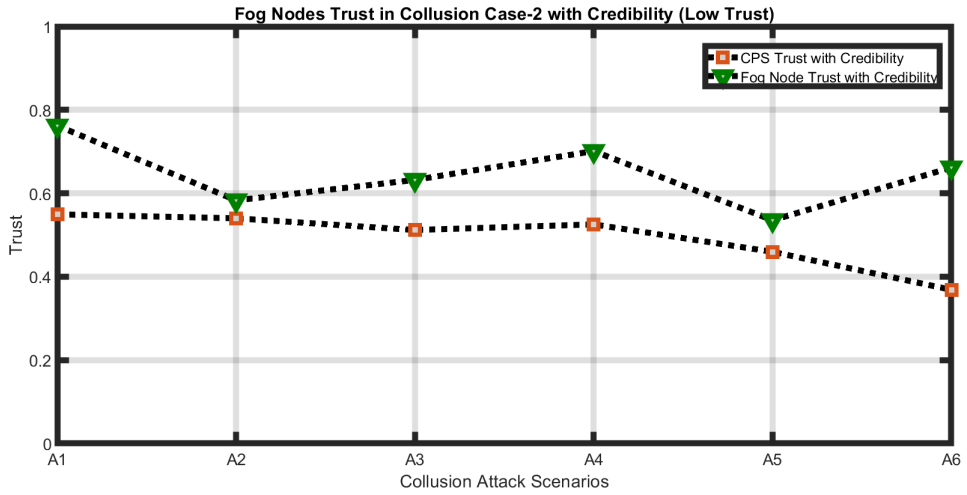


Fig. 4.33 Fog Nodes Low Trust with Credibility in Hostile Environment (***Second*** Case $C_2$)

trust $T^{cps \to fog}$ is decreasing from 0.54 to 0.25 in each subsequent attacking scenario. The decrease in $T^{cps \to fog}$ also decreased the $T^{fog}$ which dropped from 0.74 in $A_1$ to 0.60 in $A_6$.

**By Considering the Credibility Model**

The trust results computed with the credibility model are shown in Figures in 4.32 and 4.33. Again, both conditions of devices increasing and decreasing the CPS trust are considered. Figure 4.32 illustrates the results with high trust. It is noted that the credibility model successfully identifies change in trust in consecutive time instances and/or attacking scenarios and adjusts it accordingly. Overall the CPS trust $T^{cps \to fog}$ remained between 0.54 and 0.56. As a result, $T^{fog}$ also did not decrease much. However, the change in $T^{fog}$ is due to the objective trust $T^{fa \to fog}$ in the specific time instance.

Figure 4.33 shows the CPS and fog node trust when CPS devices are colluding to decrease the trust of fog nodes. The CPS trust is decreasing as more and more devices are sending values between 0.5 and 0.1 in different attacking scenarios. As a result of this, the CPS trust decreased from 0.54 to 0.36. It is underlined that CPS trust without credibility reached 0.25 in $A_6$ (see Figure 4.33) , however due to the credibility model it did not change so low this time. Moreover, due to the credibility model the fog node trust $T^{fog}$ remained between 0.76 and 0.67.

### 4.2.8 Resilience against Attacks

The Fog-CPS systems can be vulnerable to many attacks as explained in section 1.2, with the aim of attackers to degrade the accuracy of trust computation model or impact the availability

of the TMS . For example, in collusion attack, the malicious attackers can collaborate together to increase/decrease the trust of fog nodes. Likewise, in self-promoting and bad-mouthing attacks, the compromised devices report positive and negative parameters to change the trust of fog nodes.

It can be observed that despite having different nature of attacks, in all cases, trust changes dramatically and therefore the key to addressing this challenge was to detect and subsequently adjust the change in trust. In line with this, a notion of trust credibility evaluation was introduced and the change in trust is quantified by correlating it with the standard deviation. We believe that the adoption of a generalized technique i.e. measuring standard deviation of trust in hostile and hostile free environments is adequate to develop a resilient trust management system. The proposed approach is also similar to the credibility model proposed in Noor et al. (2016) which countermeasures the Sybil and collusion attacks.

### 4.2.9 Comparative Analysis

As the research in fog computing is in its early stages, there are very few trust models. To the best of this researcher's knowledge, none of the existing trust models adopt such a holistic approach for trust computation in a Fog-CPS systems. Another advantage of the proposed model is that trust is computed for both fog nodes and CPS devices. There is no trust model which computes trust for both fog nodes and CPS devices. Due to these limitations, the fog node trust $T^{fog}$ results of the proposed model are not comparable to existing approaches. However, the CPS trust results are compared with one existing study Namal et al. (2015).

#### Autonomic Trust Management Framework Namal et al. (2015)

In this work, a trust model for dynamic cloud based IoT systems is proposed. The autonomic trust management framework is based on IBM 's MAPE-K feedback control loop. The systems model consists of three layers, service consumer layer, cloud network layer, and applications and service layer. Furthermore, the service consumer layer consists of open Application Programmable Interfaces (APIs) on which clients access the services and trust agents that locally filter trust related information to the trust data pool. In the second layer, the cloud network is implemented with the service ("TaaS") which utilizes cloud based computing intelligence to obtain the corresponding parameters. These parameters are then fed to the MAPE-K feedback control loop that produces the set of trust parameters on which the final decision is made. However, the process runs over many iterations to modify a final result based on the past history. The model is evaluated by quantifying four trust parameters, namely availability, reliability, response time and capacity. However, a major limitation of the

proposed framework is the inability to detect the data anomalies. With the current proposed model, if the anomalous parameters are incorporated into trust computation then the resulting CPS trust is inaccurate and does not fall between -1 and 1, as reported in Equations (1) and (2) in Section 6 of Namal et al. (2015). However, in the proposed trust computation model, the data anomalies are detected. Moreover, all parameter values which are out of the range are not considered in trust computation. In order to evaluate the limitations of autonomic trust management system three cases of comparison are considered:

1. Normal case
2. Parameters values greater than $V_{max}$ (upper limit of range)
3. Parameter values less than $V_{min}$ (lower limit of range)

In all of the above cases, the CPS trust $T^{cps \rightarrow fog}$ computed by the proposed model is compared with Namal et al. (2015) . Moreover, in each case, 10 samples are taken for comparison. These are randomly taken samples and do not belong to a specific attacking scenario and time instance.

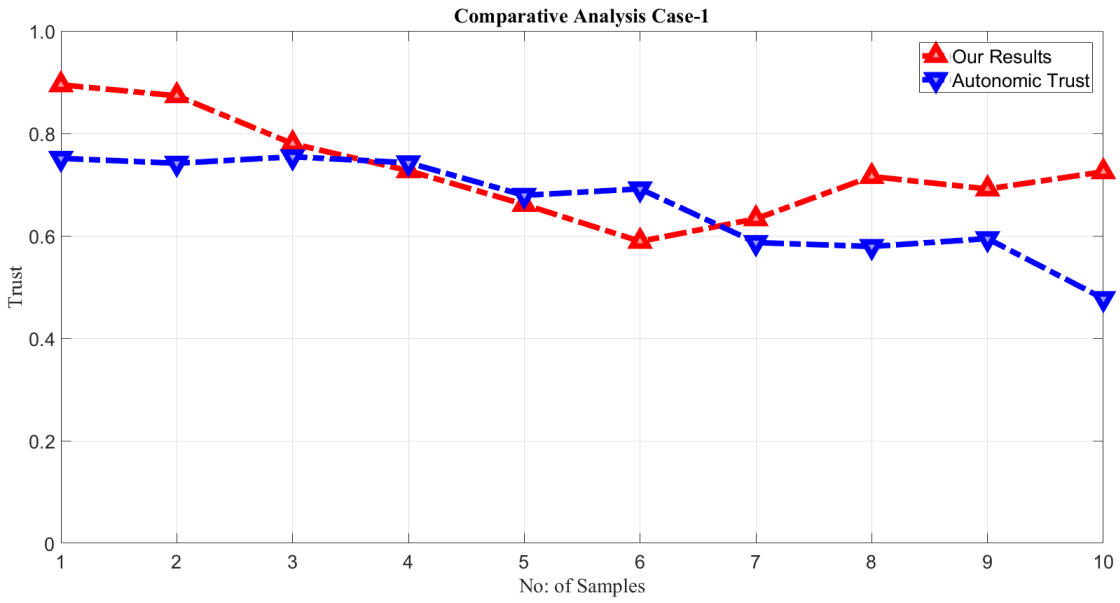


Fig. 4.34 Comparative Analysis Normal Case

### 1. Comparative Analysis - Normal Case

In the normal case, it is assumed that all parameters are within a given range, i.e. between $V_{min}$ and $V_{max}$ values introduced in Equation 2. $V_{min}$ and $V_{max}$ represent the minimum and

maximum raw data reported by an IoT device. Subsequently, the result of normal case are compared with the proposed CPS trust model. Figure 4.34 illustrates the CPS trust $T^{cps \rightarrow fog}$ of both models. It can be seen that the CPS trust computed by the proposed model lies between 0.58 and 0.89. Similarly, in case of the autonomic trust model $T^{cps \rightarrow fog}$ lies between 0.47 and 0.75. Overall, in the normal case, the trust computation in both models is trustworthy.

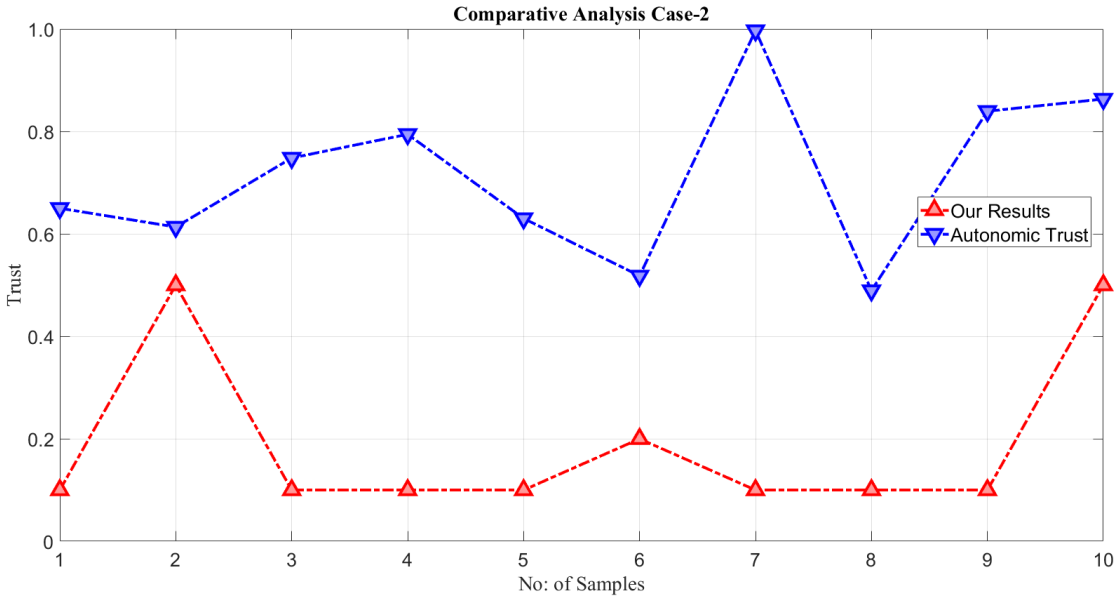


Fig. 4.35 Comparative Analysis Case-2

### 2. Comparative Analysis Case-2 $(> V_{max})$

In the second case, the parameter values greater than $V_{max}$ are considered. Figure 4.35 shows the CPS trust $T^{cps \rightarrow fog}$ results. It can be analyzed that CPS trust computed by the proposed model lies between 0.1 and 0.5. However, in case of autonomic model, the CPS trust lies between 0.48 to 0.99. However, there are false parameters but the autonomic trust model is considering them therefore resulting in inaccurate trust results.

### 3. Comparative Analysis $(< V_{min})$

In the third case, the parameter values less than $V_{min}$ are considered. The next step is to normalize the parameter values using Equation 2. However, interestingly, the normalized parameters are resulting into values greater than 1. It is contrary to -1 and 1 range reported in the paper.

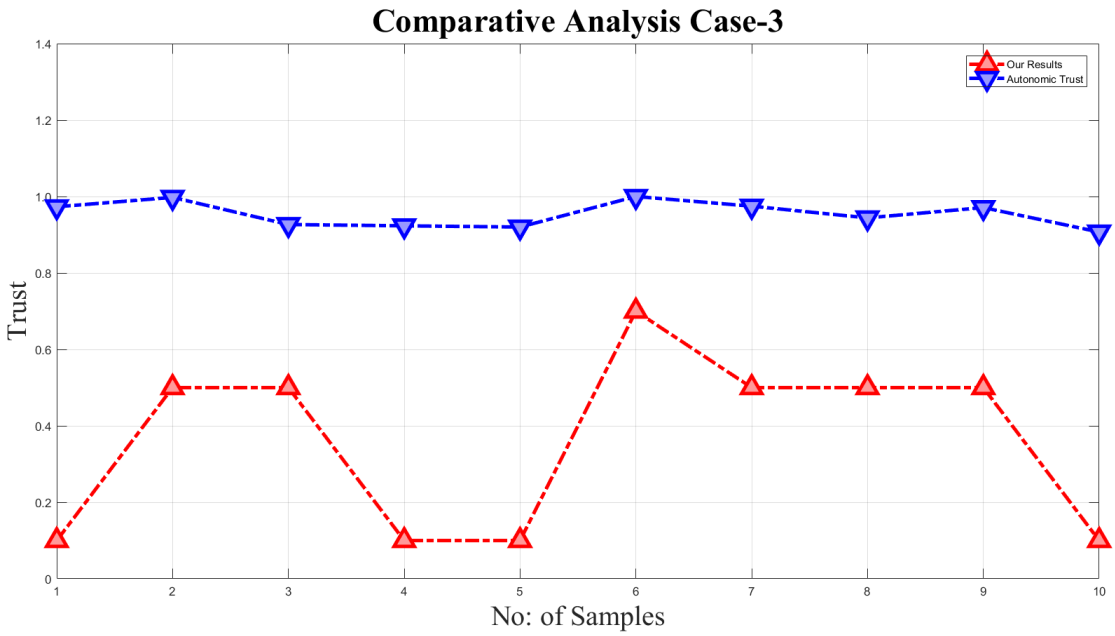


Fig. 4.36 Comparative Analysis Case-3

Figure 4.36 illustrates the CPS trust $T^{cps \rightarrow fog}$ values in case-3. However, due to other parameters and weight assignment, the CPS trust is in the given range. It can be observed in Figure 4.36 that the proposed model can detect the data anomalies and therefore compute accurate CPS trust. However, in case of autonomic trust, all trust values are equal to 1 which is not a correct trust quantification.

Overall it is maintained that the normalization introduced in Namal et al. (2015) does not take into consideration the parameter reports sent by compromised CPS devices. The anomalous parameter values do not compute an accurate and precise trust scores. However, the proposed trust credibility evaluation model takes care of all these aspects and therefore computes CPS trust with an improved accuracy and precision. Moreover, in the above comparative analysis, only one parameter is changed with respect to $V_{min}$ and $V_{max}$ and no collusion attacks are considered as well. Multiple parameters and collusion attacks can further affect the trust results.

### 4.2.10 Trust Computation Processing Time

#### Raspberry Pi

Figure 4.38 reports the overheads of the different trust results namely, objective trust $T^{fa \rightarrow fog}$, CPS Device trust $T^{cps \rightarrow fog}$, trust credibility evaluation and fog node trust $T^{fog}$. It is noted

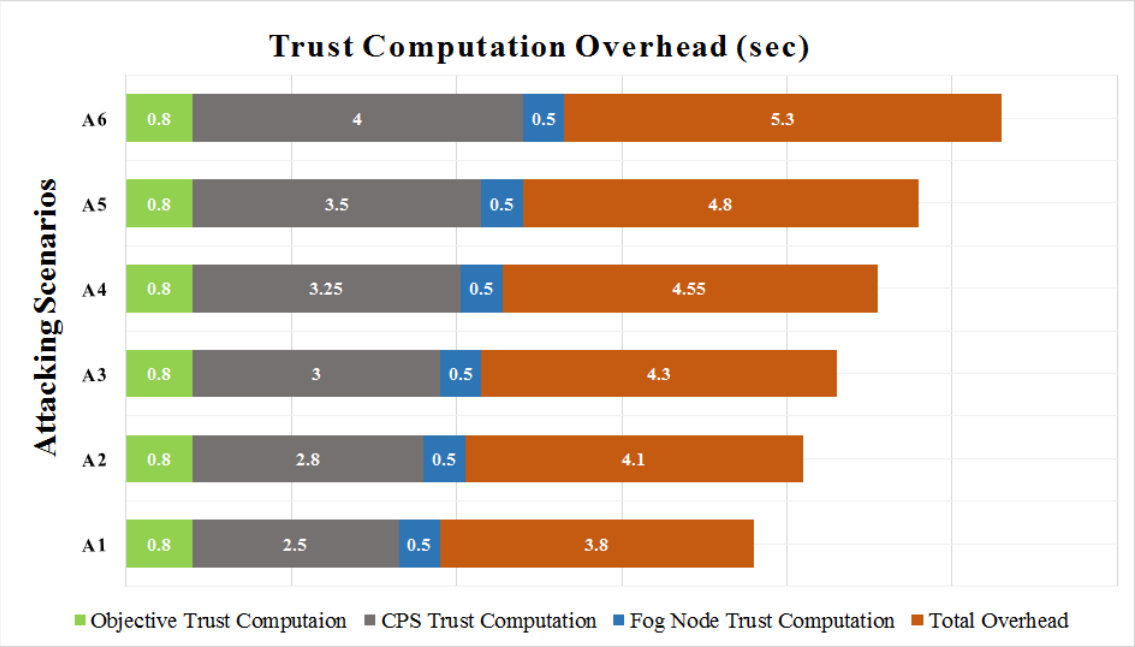


Fig. 4.37 Trust Computation Overhead

that the random forest regression executed as part of the objective trust $T^{fa \to fog}$ computation took only 2.23 seconds. Fog node $T^{fog}$ trust calculation took 0.72 seconds. However, the CPS device trust computation $T^{cps \to fog}$ took 3.98 seconds. Time taken by $T^{cps \to fog}$ is the summation of time required for data normalization, random forest regression training and testing. Additionally, the trust credibility evaluation took just 0.27 seconds. Lastly, the entire trust computation is done in 7.2 seconds. The above results demonstrate that the proposed TMS is lightweight and incurs small computation overhead even on a Raspberry Pi 3. Above results demonstrate that trust can easily be computed on resource limited devices by incurring very less overhead.

### Desktop Computer

Figure 4.37 reports the overheads of the objective trust $T^{fa \to fog}$, CPS trust $T^{cps \to fog}$ and fog node trust $T^{fog}$ in all attacking scenarios from $A_1$ to $A_6$. It is noted that the overhead of objective trust $T^{fa \to fog}$ and fog node $T^{fog}$ in all attacking scenarios is 0.8 and 0.5 seconds respectively. However, the CPS $T^{cps \to fog}$ trust computations are incurring different overheads in each attacking scenario. The CPS trust computation overhead is the summation of time required for data anomalies detection, random forest regression training and testing and trust credibility evaluation. The timings are different because the random forest regression training and testing takes a different amount of time in each attacking scenario.

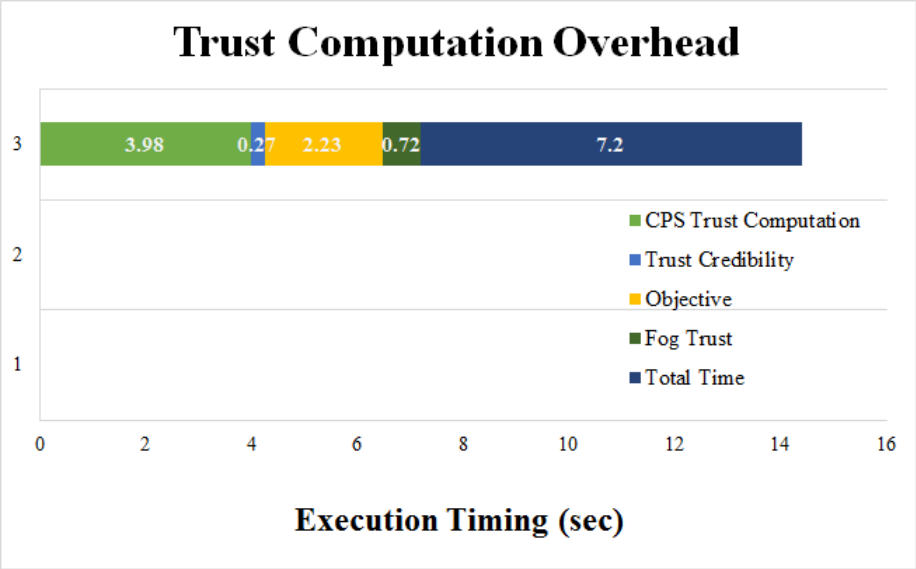


Fig. 4.38 Trust Computation Overhead

Precisely, $T^{cps \rightarrow fog}$ takes 2.5 sec in $A_1$, 2.8 sec in $A_2$, 3 sec in $A_3$, 3.25 sec in $A_4$, 3.5 sec in $A_5$, and 4 sec in $A_6$. Overall, trust computation in all attacking scenarios $[A_1, A_2, \ldots, A_6]$ takes 3.8, 4.1, 4.3, 4.5, 4.8, and 5.3 seconds respectively. The above results demonstrate that the proposed TMS is lightweight and incurs small computation overhead, hence it is suitable for large scale and dynamic Fog-CPS systems.

## 4.3 Concluding Remarks

Having presented the results of the proposed secure integrated framework components and their integration, some observations made regarding the results and a few lessons learned are discussed.

The proposed Fog-CPS scheme is compared with five other constant-size ciphertext and key schemes ECC-CPABE (Odelu and Das, 2016), CPABE14 (Guo et al., 2014), CPABE14 (Guo et al., 2014), PP-CPABE15 (Zhou et al., 2015), CPABE11 (Chen et al., 2011) and COM-CPABE14 (Shota Yamada and Kunihiro, 2014) because such schemes are computationally efficient and require fewer group elements than other ABE schemes in which the size of keys and ciphertext is dependent on the number of attributes in access policy and user attribute set. It is therefore rational to compare the proposed scheme with constant-size ABE schemes. The scheme was also compared with an asymmetric public key scheme based on elliptic curve cryptography. The performance of the proposed scheme is evaluated by analysing

timing results, memory and computational overhead. Obtaining these results was essential to compare different schemes and analyze the effectiveness of the proposed scheme.

Moreover, the proposed key update algorithms are very lightweight as each key update only incurs the overhead of one extra key component. However, if the attributes of CPS devices have been modified or changed for any reason, then Algorithms 6, 7 and 8 will need to make computation for $t \pm 1$ attributes. In such circumstances, their computational and memory complexity is similar to Algorithms 1, 2 and 3. The proposed scheme has a major advantage over all other schemes because it is based on the elliptic curve group rather than a bilinear group.

The proposed Fog-CPS scheme is lightweight compared to other schemes, however, the execution timings of encryption and decryption algorithms are comparable to other schemes based on bilinear pairing. The encryption and decryption algorithms are taking quite a lot of processing time. It is believed that that these algorithms could be made more efficient by either precomputing the ciphertext components and/or finding more new and/novel methods to compute the polynomials. Additionally, other methods to compute the scalar multiplication in elliptic curves can be employed to encrypt and decrypt the messages more efficiently. Lastly, the performance of encryption and decryption algorithms must be evaluated over different benchmarks to find their upper bounds.

Likewise, the performance of the proposed TMS is also evaluated. The trustworthiness of CPS devices and fog nodes is evaluated based on QoS and network communication features by employing the random forest regression model. Moreover, considering the possible deployment of Fog-CPS systems in hostile and unprotected environments and the compromise of the CPS devices and fog nodes, the credibility of trust is evaluated. A credibility evaluation model is designed to countermeasure the malicious behaviour (i.e. collusion, Sybil, self-promotion and bad-mouthing) of compromised entities.

In current experimental results, the credibility model is evaluated to countermeasure only the collusion attacks. The results show that the collusion attacks are successfully prevented and subsequently their effect is mitigated by the proposed credibility model. But as mentioned above, the Fog-CPS system can be vulnerable to other attacks so it would be interesting to evaluate the credibility model in all cases to ascertain the claim. In future work, the effectiveness of the credibility model can be evaluated under different attacking scenarios. Furthermore, the standard deviation $\acute{\sigma}$ in the credibility model is calculated by taking into consideration two hostile environments, it would be interesting to calculate the standard deviation $\acute{\sigma}$ under different network configurations.

Additionally, the experimental results are compared with an existing model Namal et al. (2015) which cannot detect and therefore prevent the data anomalies attack. The results

demonstrate that the proposed TMS can not only detect the data anomalies but also prevent other malicious behaviours of compromised entities.

The trust computation timing over a resource constrained device Raspberry PI 3 and a normal desktop computer are measured to analyze the performance efficiency of the proposed TMS for different Fog-CPS entities. The results show that the proposed TMS is lightweight and fast. The integration of the SC and TMS is presented by elaborating how the change in trust of a Fog-CPS entity triggers a change in its access control rights. Lastly, considering the recent endeavours to urbanization, more specifically the research in smart cities, the TMS proposal is timely and important.

## 4.4 Summary of the Chapter

This chapter experimentally evaluates the proposed framework. The SC was first evaluated in Section 4.1 where the time and space complexity of the Fog-CPS scheme was elaborated. Additionally, the TMS was evaluated with and the results presented in Section 4.2. The results demonstrate that the proposed framework is lightweight, efficient and accurate. It can address the security and trust challenges of the Fog-CPS systems.

# Chapter 5

# Conclusions and Future Work

This chapter concludes the author's work by revisiting the thesis goals, contributions and achieved objectives. This includes the author's work i.e. the proposed secure integrated framework. Lastly, the author discusses possible future directions based on the research performed by the author after systematically reviewing the work related to Fog-CPS systems.

## 5.1 Restating Research Problems and Research Goals

As an emerging paradigm, Fog-CPS systems combine computation and communication capabilities with the physical space. Fog-CPS systems can improve the monitoring and management of next generation computer systems. Despite the opportunities provided by Fog-CPS, they face increased security and trust challenges.

To overcome these challenges, a secure integrated framework comprised of a security component (SC) and a trust management system (TMS) component was proposed. As part of the SC, a lightweight encryption scheme is proposed to establish identity and access control management. Furthermore, a trust computation model is proposed to evalute the trustworthiness of Fog-CPS entities. Discussion now moves to the research objectives and how are they achieved.

1. **Objective 1 - To investigate the security and trust challenges of Fog-CPS systems.**

   In Chapter 2, the author reviewed the most recent and cited studies proposed for Fog-CPS systems, CPS system, cloud-enabled IoT systems and fog computing. Existing works are studied with the aim to investigate the security and trust challenges.

2. **Objective 2 - To integrate security and trust into a framework for the Fog-CPS systems.**

In Chapter 3, the author proposed a secure integrated framework having security and trust as the fundamental components.

3. **Objective 3 - To propose an efficient and lightweight security component which addresses security challenges namely data confidentiality, authentication, and authorization.**

   In Chapter 3, the author designed a new lightweight encryption scheme based on elliptic curve cryptography to overcome the security challenges of Fog-CPS systems.

4. **Objective 4 - To design a resilient and accurate trust management system (TMS) for Fog-CPS systems.**

   Compared to existing trust models, the author proposed a holistic TMS that quantifies the trust for all entities in Fog-CPS systems. Chapter 3 presented the design of the TMS. The proposed TMS can countermeasure various attacks, namely collusion, on-off and bad mouthing, that aim to degrade the performance of the Fog-CPS systems and/or affect the accuracy of the trust computation process itself.

## 5.2 Research Contributions and Distribution of Work

In this thesis, the author has proposed and experimentally evaluated the secure integrated framework that overcomes the security and trust challenges of Fog-CPS systems. The researcher's contributions are as follows:

### 5.2.1 A Secure Integrated Framework

The first contribution is the design of a secure integrated framework. Although, some frameworks are proposed in literature to address the security, privacy and trust challenges, these are not sufficient to countermeasure multi-dimensional problems faced by Fog-CPS for reasons underlined in Chapters 1 and 2. The proposed framework is advantageous in several ways as listed below:

- ***Firstly***, it integrates both the security and trust properties in one solution.
- ***Secondly***, it evaluates the trustworthiness of Fog-CPS entities by taking into consideration multiple security and trust parameters.
- ***Thirdly***, it countermeasures the several malicious behaviours that can disrupt the normal operation of the Fog-CPS systems.

- ***Fourthly***, the proposed framework is a generalized solution which can easily be adapted to any specific Fog-CPS use case.

### 5.2.2 Fog-CPS Scheme

The second contribution is the design of a lightweight encryption scheme. There are several public key and ABE schemes in the literature. However, the existing schemes have a number of limitations as highlighted in Chapter 1 and 2. Moreover, one of the existing ABE scheme is adapted such that the new scheme can meet the security requirements of Fog-CPS systems; and address challenges of data confidentiality, user authentication and access management. The novel aspects of the proposed scheme are listed below:

- ***Firstly***, it does not rely on any trusted authority for key generation and distribution. All CPS devices and fog nodes can themselves generate the keys. However, an initial registration with FA is mandatory.
- ***Secondly***, the attribute set is divided into two subsets namely the public and secret. The secret attributes are only shared with the FA, whilst the public attributes can be known to other entities in the system. Additionally, in the Fog-CPS scheme, one attribute set is associated with only one CPS device.
- ***Thirdly***, the proposed scheme is more lightweight than the one it is based on.
- ***Fourthly***, two new algorithms have been designed for key generation and key update/revoke. Moreover, both these new algorithms are further split into three algorithms. To be more specific, the key generation process is divided into three algorithms, namely partial key pair generation (Algorithm 1), final public key generation (Algorithm 2) and final secret key generation (Algorithm 3). Likewise, the key revocation process is also divided into three algorithms, namely, key pair revocation (Algorithm 6), final public key revoke (Algorithm 7) and final secret key revoke (Algorithm 8).

### 5.2.3 Trust Management System

The third contribution is the design of a holistic and all encompassing TMS for Fog-CPS systems. There are several trust models in the literature but none of them can be used directly due to the inherent features and distributed nature of Fog-CPS systems. The proposed TMS is novel in a three fold manner.

- ***Firstly***, it computes trust for all entities of Fog-CPS entities.

- ***Secondly***, the trust computation is formulated as a statistical regression problem, and random forest regression is employed to solve it. To the best of the author's knowledge, this is the first research work to employ regression models more specifically, the random forest regression for computing the trust in Fog-CPS systems.

- ***Thirdly***, it includes a trust credibility evaluation module that ensures accurate and precise trust computation. As a Fog-CPS system is an inherently open and distributed system, it is vulnerable to collusion, self-promotion, bad-mouthing, ballot-stuffing, and opportunistic service attacks. The compromised CPS devices can collaborate to increase/decrease the trust of fog nodes.

  These challenges are addressed by evaluating the credibility of trust, and subsequently adjusting the trust by correlating it with a standard deviation threshold. It also verifies that the parameters sent by the CPS devices and fog nodes are within a given range i.e. not intentionally modified and/or manipulated by compromised devices. The standard deviation is quantified by comparing the expected value of trust in legitimate and hostile environments. The results demonstrate that the credibility model successfully countermeasures the malicious behaviour of compromised devices in different configurations of hostile environments. The multi-factor trust assessment and credibility evaluation enable accurate and precise trust computation and guarantee a dependable Fog-CPS.

## 5.3 Future Work

There are many directions to advance the work that has been presented in this thesis. The future work is enumerated below:

1. As a future work, the researcher aims to extend the proposed secure integrated framework such that it can also address the privacy challenges faced by the Fog-CPS systems. This extension would require adding more sub-components dealing with *privacy preferences and personalization*, *secure data search* and *big data analytics* to name a few. These sub-components could be added to the security component (SC).

2. The researcher also aims to make the Fog-CPS scheme more lightweight such that it takes fewer elliptic curve points and requires less scalar multiplications to generate the secret keys, encrypt and decrypt the data.

3. The researcher also aims to prove the security analysis in a standard model.

4. Besides that the researcher aims to incorporate the blockchain model in Fog-CPS systems. The blockchain technology is so promising and can definitely address the numerous challenges faced by these systems.

5. Furthermore, the researcher wants to quantify the user privacy in the Fog-CPS systems and intends to quantify the harm caused by privacy breaches and present the privacy statistics to concerned persons/organizations. This is very interesting future work and requires multi-disciplinary input from fields of fog computing, cyber physical systems, data privacy and law.

6. Lastly, the researcher wishes to analyze how the proposed Fog-CPS scheme could be mapped to the data privacy clauses of EU General Data Protection Regulation (GDPR) (Parliament, 2016). The GDPR demands cloud providers and/or fog provides to be in compliance with data privacy regulations before they process and store customer data. The existing privacy enhancing techniques cannot guarantee the level of privacy required by the new regulations. However, it is believed that the proposed secure integrated framework is a small step in this direction which has the potential to be a practical future technology for solving various problems faced by Fog-CPS systems including but not limited to data protection, data confidentiality, authentication, authorization and trust management.

## 5.4 Concluding Remarks

Fog-CPS systems are the future of next generation computer systems. Having been motivated by the prospects and opportunities provided by this future paradigm, the author has explored them comprehensively in a bid to identify the challenges faced by these systems. This dissertation proposed a secure integrated framework for Fog-CPS systems. The proposed framework is designed after a thorough investigation of these systems. Various dimensions (i.e. security, privacy, trustworthiness and service orchestration) of Fog-CPS systems are studied to investigate the vulnerabilities and threats faced by such complex and inherently heterogeneous systems. After identifying the security and trust challenges, efforts were made to find a solution. However, it soon became clear that Fog-CPS systems require an integrated approach which addresses all these issues simultaneously as the limitations and/or absence of one solution can be exploited by malicious attackers to disrupt these systems and impact their availability.

Considering the limitations of existing works, the researcher has designed a secure integrated framework for addressing the security and trust challenges inherent to Fog-CPS

systems. The proposed framework is comprised of two fundamental components namely the SC and TMS. The SC component ensures the security whereas the TMS guarantees the dependability of Fog-CPS entities. The identity management and access control management sub-components of SC ensure that fog nodes and CPS devices are authenticated and authorized. The TMS quantifies trust of fog nodes and CPS devices by monitoring their QoS parameters and other performance indicators. It also ensures dependable fog resources are granted to CPS devices. The researcher believes this research is timely and the proposed secure integrated framework is a step in the right direction. It can address the security and trust challenges of Fog-CPS systems.

# References

Ahmed, E. and Rehmani, M. H. (2017). Mobile edge computing: Opportunities, solutions, and challenges. *Future Generation Computer Systems*, 70:59 – 63.

Akinyele, J. A., Garman, C., Miers, I., Pagano, M. W., Rushanan, M., Green, M., and Rubin, A. D. (2013). Charm: a framework for rapidly prototyping cryptosystems. *Journal of Cryptographic Engineering*.

Al-Turjman, F., Ever, Y. K., Enver, E., X. Nguyen, H., and David, D. B. (2017). Seamless key agreement framework for mobile-sink in iot based cloud-centric secured public safety sensor networks. *IEEE Access*.

Alhanahnah, M., Bertok, P., Tari, Z., and Alouneh, S. (2018). Context-aware multifaceted trust framework for evaluating trustworthiness of cloud providers. *Future Generation Computer Systems*, 79(2):488–499.

Alippi, C. and Roveri, M. (2017). The (not) far-away path to smart cyber-physical systems: An information-centric framework. *Computer*, 50(4):38–47.

Alrawais, A., Alhothaily, A., Hu, C., Xing, X., and Cheng, X. (2017). An attribute-based encryption scheme to secure fog communications. *IEEE Access*, 5:9131–9138.

Andy, G. (2015). This gadget hacks gm cars to locate, unlock, and start them.

BBC News, B. (2014). Sony pictures computer system hacked in online attack.

BBC News, B. (2016). Sage software firm hit by data breach.

Bethencourt, J., Sahai, A., and Waters, B. (2007). Ciphertext-policy attribute-based encryption. In *Proceedings of the 2007 IEEE Symposium on Security and Privacy*.

Boneh, D. and Boyen, X. (2004). Efficient selective-id secure identity based encryption-without random oracles. In Cachin, C. and Camenisch, J., editors, *34th Annual International International Conference on Cryptographic Techniques (EUROCRYPT2004)*, Berlin. Springer-Verlag.

Boneh, D., Sahai, A., and Waters, B., editors (2011). *Proceedings of the 8th Conference on Theory of Cryptography*.

Bonomi, F., Milito, R., Zhu, J., and Addepalli, S. (2012). Fog computing and its role in the internet of things. In *Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing*, MCC '12, pages 13–16, New York, NY, USA. ACM.

Camhi, J. (2015). Former cisco ceo john chambers predicts 500 billion connected devices by 2025.

Cha, S.-C., Chuang, M.-S., Yeh, K.-H., Huang, Z.-J., and Su, C. (2018). A user-friendly privacy framework for users to achieve consents with nearby ble devices. *IEEE Access*.

Challa, S., Wazid, M., Kumar Das, A., Kumar, N., Goutham Reddy, A., Yoon, E.-J., and Yoo, K.-Y. (2017). Secure signature-based authenticated key establishment scheme for future iot applications. *IEEE Access*, 5:3028–3043.

Chen, C., Chen, J., Lim, H. W., Zhang, Z., Feng, D., Ling, S., and Wang, H. (2013). Fully secure attribute-based systems with short ciphertexts/signatures and threshold access structures. In *Proceedings of the 13th International Conference on Topics in Cryptology*.

Chen, C., Zhang, Z., and Feng, D. (2011). *Efficient Ciphertext Policy Attribute-Based Encryption with Constant-Size Ciphertext and Constant Computation-Cost*, pages 84–101. Springer Berlin Heidelberg.

Chen, J. and Zhu, Q. (2017). Security as a service for cloud-enabled internet of controlled things under advanced persistent threats: A contract design approach. *IEEE Transactions on Information Forensics and Security*, 12(11):2736–2750.

Chen, S., Li, M., Ren, K., and Qiao, C. (2015a). Crowd map: Accurate reconstruction of indoor floor plans from crowdsourced sensor-rich videos. In *2015 IEEE 35th International Conference on Distributed Computing Systems*, pages 1–10.

Chen, S., Ma, M., and Luo, Z. (2015b). An authentication framework for multi-domain machine-to-machine communication in cyber-physical systems. In editor, editor, *2015 IEEE Globecom Workshops (GC Wkshps)*.

Chen, X. and Wang, L. (2017). A cloud-based trust management framework for vehicular social networks. *IEEE Access*, 5:2967–2980.

Chiang, M. (2015). Fog networking: An overview on research opportunities. Technical report, Princeton University.

Cisco (2018). Cisco global cloud index: Forecast and methodology, 2016–2021, white paper. Technical report, Cisco.

Criminisi, A., Shotton, J., and Konukoglu, E. (2012). *Decision Forests: A Unified Framework for Classification, Regression, Density Estimation, Manifold Learning and Semi-Supervised Learning*, volume 7. Now Publishers.

De Caro, A., Iovino, V., and Persiano, G., editors (2013). *Fully Secure Hidden Vector Encryption*.

Diffie, W. and Hellman, M. (1976). New directions in cryptography. *IEEE Transactions Information Theory*, 22(6).

DiMase, D., Collier, Z. A., Heffner, K., and Linkov, I. (2015). Systems engineering framework for cyber physical security and resilience. *Environment Systems and Decisions*, 35(2):291–300.

Diro, A. A., Chilamkurti, N., and Nam, Y. (2018). Analysis of lightweight encryption scheme for fog-to-things communication. *IEEE Access*, 6:26820–26830.

Emura, K., Miyaji, A., Nomura, A., Omote, K., and Soshi, M. (2009). *A Ciphertext-Policy Attribute-Based Encryption Scheme with Constant Ciphertext Length*, pages 13–23. Springer Berlin Heidelberg.

Evans, D. (2011). The internet of things how the next evolution of the internet is changing everything. Technical report, Cisco Internet Business Solutions Group (IBSG).

Fan, W. and Perros, H. (2014). A novel trust management framework for multi-cloud environments based on trust service providers. *Knowledge-Based Systems*.

Ferrer, A. J., HernáNdez, F., Tordsson, J., Elmroth, E., Ali-Eldin, A., Zsigri, C., Sirvent, R., Guitart, J., Badia, R. M., Djemame, K., Ziegler, W., Dimitrakos, T., Nair, S. K., Kousiouris, G., Konstanteli, K., Varvarigou, T., Hudzia, B., Kipp, A., Wesner, S., Corrales, M., Forgó, N., Sharif, T., and Sheridan, C. (2012). Optimis: A holistic approach to cloud service provisioning. *Future Generation Computer Systems*, 28(1):66–77.

Freeman, D., Scott, M., and Teske, E. (2009). A taxonomy of pairing-friendly elliptic curves. *Journal of Cryptology*, 23(2):224–280.

Freeman, D., Scott, M., and Teske, E. (2010). A taxonomy of pairing-friendly elliptic curves. *Journal of Cryptology*, 23(2).

Galbraith, S. D., Paterson, K. G., and Smart, N. P. (2008). Pairings for cryptographers. *Discrete Applied Mathematics*, 156(16):3113–3121.

Gallagher, P. (October, 2012). Federal information processing standard (fips) 180-4. Technical report, NIST.

Gazis, V. (2017). A survey of standards for machine-to-machine and the internet of things. *IEEE Communications Surveys Tutorials*, 19(1):482–511.

George, G. and M. Thampi, S. (2018). A graph-based security framework for securing industrial iot networks from vulnerability exploitations. *IEEE Access*.

Ghena, B., Beyer, W., Hillaker, A., Pevarnek, J., and Halderman, J. A. (2014). Green lights forever: Analyzing the security of traffic infrastructure. In *Proceedings of the 8th USENIX Conference on Offensive Technologies*, WOOT'14, pages 7–7, Berkeley, CA, USA. USENIX Association.

Ghosh, N., Ghosh, S. K., and Das, S. K. (2015). Selcsp: A framework to facilitate selection of cloud service providers. *IEEE Transactions on Cloud Computing*, 3(1):66–79.

GNU (2019). Gnu octave scientific programming language, https://www.gnu.org/software/octave/.

Goyal, V., Pandey, O., Sahai, A., and Waters, B. (2006). Attribute-based encryption for fine-grained access control of encrypted data. In *Proceedings of the 13th ACM Conference on Computer and Communications Security*, pages 89–98. ACM, Alexandria, Virginia, USA.

Guo, F., Mu, Y., Susilo, W., Wong, D. S., and Varadharajan, V. (2014). Cp-abe with constant-size keys for lightweight devices. *IEEE Transactions on Information Forensics and Security*.

Guo, J., Chen, I.-R., and Tsai, J. J. (2017). A survey of trust computation models for service management in internet of things systems. *Computer Communications, Elsevier*, pages 1–14.

Gupta, H., Dastjerdi, A. V., Ghosh, S. K., and Buyya, R. (2017). ifogsim: A toolkit for modeling and simulation of resource management techniques in the internet of things, edge and fog computing environments. *Cloud and Fog Computing, Wiley Online Library*, pages 1275–1296.

Habib, S. M., Varadharajan, V., and Mühlhäuser, M., editors (2013). *A Trust-Aware Framework for Evaluating Security Controls of Service Providers in Cloud Marketplaces*.

Hahn, A., Thomas, R. K., Lozano, I., and Cardenas, A. (2015). A multi-layered and kill-chain based security analysis framework for cyber-physical systems. *International Journal of Critical Infrastructure Protection*, 11:39 – 50.

Han, Z., Li, X., Huang, K., and Feng, Z. (2018). A software defined network-based security assessment framework for cloudiot. *IEEE Internet of Things Journal*, 5(3):1424–1434.

Herranz, J. (2014). Attribute-based signatures from rsa. *Theoretical Computer Science*.

Hess, F., Smart, N. P., and Vercauteren, F. (2006). The eta pairing revisited. *IEEE Transactions on Information Theory*, 52(10):4595–4602.

Hevia, A. (2013). Introduction to provable security. In *Advanced Crypto School, Florianpolis*.

Hu, C., Li, H., Huo, Y., , and Liao, X. (2016). Secure and efficient data communication protocol for wireless body area networks. *IEEE Transactions on Multi-Scale Computing Systems*, 2(2).

Hu, P., Ning, H., Qiu, T., Song, H., Wang, Y., and Yao, X. (2017). Security and privacy preservation scheme of face identification and resolution framework using fog computing in internet of things. *IEEE Internet of Things Journal*, 4(5):1143–1155.

Ian, F. B., Gadiel, S., and Nigel, P. S. (1999). *Elliptic Curves in Cryptography (London Mathematical Society Lecture Note Series)*. Cambridge University Press, Cambridge, UK.

Inc., C. S. (2015). Fog computing and the internet of things: Extend the cloud to where the things are, white paper. Technical report, Cisco, San Jose, CA, USA.

Jalali, F., Hinton, K., Ayre, R., Alpcan, T., and Tucker, R. S. (2016). Fog computing may help to save energy in cloud computing. *IEEE Journal on Selected Areas in Communications*, 34(5):1728–1739.

Jiang, Y., Susilo, W., Mu, Y., and Guo, F. (2018). Ciphertext-policy attribute-based encryption against key-delegation abuse in fog computing. *Future Generation Computer Systems*, 78(P2):720–729.

Josang, A. and Ismail, R. (2007). A survey of trust and reputation systems for online service provision. *Decision Support Systems*, 42(3):618– 644.

Joux, A., editor (2000). *A one round protocol for tripartite Diffie-Hellman*.

Julio, L. and Ricardo, D. (2000). An overview of elliptic curve cryptography. Technical report, University of Campinas.

Khalid, A., Kirisci, P., Khan, Z. H., Ghrairi, Z., Thoben, K.-D., and Pannek, J. (2018). Security framework for industrial collaborative robotic cyber-physical systems. *Computers in Industry*, 97:132 – 145.

Khan, Z. U. A. (2015). *Efficient Design and Implementation of Elliptic Curve Cryptography on FPGA*. PhD thesis, Department of Electronic and Electrical Engineering, The University of Sheffield.

Kim, H., Kang, E., Lee, E. A., and Broman, D. (2017). A toolkit for construction of authorization service infrastructure for the internet of things. In *Proceedings of the Second International Conference on Internet-of-Things Design and Implementation*, IoTDI '17, pages 147–158, New York, NY, USA. ACM.

Kim, H. and Lee, E. A. (2017). Authentication and authorization for the internet of things. *IT Professional*, 19(5):27–33.

Kim, H., Lee, H., Kim, W., and Kim, Y. (2010). A trust evaluation model for qos guarantees in cloud systems. *International Journal of Grid and Distributed Computing*, 3(1).

Kim, H., Wasicek, A., Mehne, B., and A. Lee, E. (2016a). A secure network architecture for the internet of things based on local authorization entities. In *2016 IEEE 4th International Conference on Future Internet of Things and Cloud (FiCloud)*, pages 114–122.

Kim, Y.-J., Kolesnikov, V., and Thottan, M. (2016b). Resilient end-to-end message protection for cyber-physical system communications. *IEEE Transactions on Smart Grid*, PP.

Koblitz, N. (1987). Elliptic curve cryptosystems. *Mathematics of Computations*, 48(177):203–209.

Koblitz, N., Menezes, A., and Vanstone, S. (2000). The state of elliptic curve cryptography. *Designs, Codes and Cryptography*, 19(2):173–193.

Kocabas, O., Soyata, T., and Aktas, M. K. (2016). Emerging security mechanisms for medical cyber physical systems. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*.

Komninos, N. and Junejo, A. K. (2015). Privacy preserving attribute based encryption for multiple cloud collaborative environment. In *2015 IEEE/ACM 8th International Conference on Utility and Cloud Computing (UCC)*, pages 595–600.

Krebs, B. (2014). Banks: Credit card breach at home depot.

Kushner, D. (2013). The real story of stuxnet. *IEEE Spectrum*, 50(3):48–53.

Lee, E., Lee, H.-S., and Park, C.-M. (2009). Efficient and generalized pairing computation on abelian varieties. *IEEE Transactions on Information Theory*, 55(4):1793–1803.

Lee, R. M., Assante, M. J., and Conway, T. (2016). Analysis of the cyber attack on the ukrainian power grid. Technical report, SANS ICS.

Lenstra, A. K. and Verheul, E. R. (2001). Selecting cryptographic key sizes. *Journal of Cryptology*, 14(4):255–293.

Lewis, D. (2014). icloud data breach: Hacking and celebrity photos.

Lewis, D. (2017). The ddos attack against dyn one year later.

Lewko, A., Okamoto, T., Sahai, A., Takashima, K., and Waters, B., editors (2010). *Fully Secure Functional Encryption: Attribute-based Encryption and (Hierarchical) Inner Product Encryption*.

Lewko, A. and Waters, B., editors (2011). *Decentralizing Attribute-based Encryption*.

Li, B., Liao, L., Leung, H., and Song, R. (2014). Phat: A preference and honesty aware trust model for web services. *IEEE Transactions on Network and Service Management*, 11(3):363–375.

Li, D., Aung, Z., Williams, J., and Sanchez, A. (2014). P3: Privacy preservation protocol for automatic appliance control application in smart grid. *IEEE Internet of Things Journal*, 1(5):414–429.

Li, L., Gu, T., Chang, L., Xu, Z., Liu, Y., and Qian, J. (2017). A ciphertext-policy attribute-based encryption based on an ordered binary decision diagram. *IEEE Access*.

Li, W. and Song, H. (2017). Art: An attack-resistant trust management scheme for securing vehicular ad hoc networks. *IEEE Transactions on Intelligent Transportation Systems*, 17(4):960–969.

Li, X., Ma, H., Zhou, F., and Gui, X. (2015). Service operator-aware trust scheme for resource matchmaking across multiple clouds. *IEEE Transactions on Parallel and Distributed Systems*, 26(5):1419–1429.

Lu, L. and Yuan, Y. (2018). A novel topsis evaluation scheme for cloud service trustworthiness combining objective and subjective aspects. *Journal of Systems and Software*, 143:71–86.

Lynn, B. (2007a). *On the Implementation of Pairing-based Cryptosystems*. PhD thesis, Stanford University.

Lynn, B. (2007b). The stanford pairing based crypto library. http://crypto.stanford.edu/pbc.

Mahmood, K., Chaudhry, S. A., Naqvi, H., Kumari, Saru, L. X., and Sangaiah, A. K. (2018). An elliptic curve cryptography based lightweight authentication scheme for smart grid communication. *Future Generation Computer Systems*, 81(1).

Maji, H. K., Prabhakaran, M., and Rosulek, M. (2011). Attribute-based signatures. In *Proceedings of the 11th International Conference on Topics in Cryptology: CT-RSA 2011*.

Matsuda, S., Kanayama, N., Hess, F., and Okamoto, E. (2007). Optimised versions of the ate and twisted ate pairings. In Galbraith, S. D., editor, *Cryptography and Coding*, pages 302–312, Berlin, Heidelberg. Springer Berlin Heidelberg.

Mayer, S., Hodges, J., Yu, D., Kritzler, M., and Michahelles, F. (2017). An open semantic framework for the industrial internet of things. *IEEE Intelligent Systems*, 32(1):96–101.

Menezes, A. J., Vanstone, S. A., and Oorschot, P. C. V. (1996). *Handbook of Applied Cryptography*. CRC Press, Inc., Boca Raton, FL, USA, 1st edition.

Mick, T., Tourani, R., and Misra, S. (2018). Laser: Lightweight authentication and secured routing for ndn iot in smart cities. *IEEE Internet of Things Journal*, 5(2).

Miller, V. S. (1986). Use of elliptic curves in cryptography. In Williams, H. C., editor, *Advances in Cryptology — CRYPTO '85 Proceedings*, pages 417–426, Berlin, Heidelberg. Springer Berlin Heidelberg.

Miller, V. S. (2004). The weil pairing, and its efficient calculation. *Journal of Cryptology*, 17(4):235–261.

Mohsin, M., Sardar, M. U., Hasan, O., and Anwar, Z. (2017). Iotriskanalyzer: A probabilistic model checking based framework for formal risk analytics of the internet of things. *IEEE Access*, 5:5494–5505.

Montgomery, P. L. (1987). Speeding the pollard and elliptic curve methods of factorization. *Mathematics of Computations*, 48(177):243–264.

Moody, D., Peralta, R., Perlner, R., Regenscheid, A., Roginsky, A., and Chen, L. (2015). Report on pairing-based cryptography. *Journal of Research of the National Institute of Standards and Technology*, 120(002).

Morris, J. D. (August, 2015). Federal information processing standard (fips) 202. Technical report, NIST.

Muhammad, K., Hamza, R., Ahmad, J., Lloret, J., Wang, H., and Wook Baik, S. (2018). Secure surveillance framework for iot systems using probabilistic image encryption. *IEEE Transactions on Industrial Informatics*, 14(8).

Nagarajan, A. and Varadharajan, V. (2011). Dynamic trust enhanced security model for trusted platform based services. *Future Generation Computer Systems*, 27(5).

Nair, S. K., P. S. D. T. F. A. J. T. J. S. T. S. C. R. M. and Khan, A. U. (2010). Towards secure cloud bursting, brokerage and aggregation. In editor, editor, *IEEE European conference on Web Services*.

Namal, S., Hasindu, G., Myoung Lee, G., and Um, T.-W. (2015). Autonomic trust management in cloud-based and highly dynamic iot applications. In *ITU Kaleidoscope: Trust in the Information Society (K-2015)*, Barcelona, Spain. IEEE.

Ni, J., Lin, X., Zhang, K., and Shen, X. (2016). Privacy-preserving real-time navigation system using vehicular crowdsourcing. In *2016 IEEE 84th Vehicular Technology Conference (VTC-Fall)*, pages 1–5.

Ni, J., Zhang, K., Lin, X., and Shen, X. S. (2018). Securing fog computing for internet of things applications: Challenges and solutions. *IEEE Communications Surveys & Tutorials*, 20(1):601–628.

NIST (2017). Framework for cyber-physical systems: Volume 1, overview. Report 1, NIST.

Nitti, M., Girau, R., and Atrozi, L. (2014). Trustworthiness management in the social internet of things. *IEEE Transactions on Knowledge and Data Engineering*, 26(5):1253 – 1266.

Noor, T. H., Sheng, Q. Z., Yao, L., Dustdar, S., and Ngu, A. H. (2016). Cloudarmor: Supporting reputation-based trust management for cloud services. *IEEE Transactions on Distributed Systems*, pages 367 – 380.

Nuttapong Attrapadung, Javier Herranz, F. L. B. L. E. d. P. C. R. (2013). Attribute-based encryption schemes with constant-size ciphertexts. *Theoretical Computer Science*.

Odelu, V. and Das, A. K. (2016). Design of a new cp-abe with constant-size secret keys for lightweight devices using elliptic curve cryptography. *Security and Communication Networks*.

OpenFog Consortium Architecture Working Group, G. (2017). Openfog reference architecture for fog computing. Technical report, OpenFog Consortium.

Oualha, N. and Nguyen, K. T. (2016). Lightweight attribute-based encryption for the internet of things. In *2016 25th International Conference on Computer Communication and Networks (ICCCN)*.

Parliament, E. (2016). General data protection regulation (gdpr.

Pawlick, J. and Zhu, Q. (2017). Strategic trust in cloud-enabled cyber-physical systems with an application to glucose control. *IEEE Transactions ON Information Forensics and Security*, 12(12):2906– 2919.

Peng, M., Yan, S., Zhang, K., and Wang, C. (2016). Fog-computing-based radio access networks: issues and challenges. *IEEE Network*, 30(4):46–53.

Phuong, T. V. X., Yang, G., and Susilo, W. (2016). Hidden ciphertext policy attribute-based encryption under standard assumptions. *IEEE Transactions on Information Forensics and Security*.

Puthal, D., Malik, N., Mohanty, S. P., Kougianos, E., and Yang, C. (2018). The blockchain as a decentralized security framework [future directions]. *IEEE Consumer Electronics Magazine*, 7(2):18–21.

Rahman, M. A., Hossain, M. S., Loukas, G., Hassanain, E., Rahman, S. S., Alhamid, M. F., and Guizani, M. (2018). Blockchain-based mobile edge computing framework for secure therapy applications. *IEEE Access*, 6:72469–72478.

Rathore, S., Sharma, P. K., Sangaiah, A. K., and Park, J. J. (2018). A hesitant fuzzy based security approach for fog and mobile-edge computing. *IEEE Access*, 6:688–701.

Rein, A., Rieke, R., Jäger, M., Kuntze, N., Coppolino, Luigi", e. A., Cuppens-Boulahia, N., Cuppens, F., Katsikas, S., and Lambrinoudakis, C. (2016). Trust establishment in cooperating cyber-physical systems. In *Security of Industrial Control Systems and Cyber Physical Systems*, pages 31–47, Cham. Springer International Publishing.

Rivest, R. (1992). The md5 message-digest algorithm. *RFC*.

Rivest, R. L., Shamir, A., and Adleman, L. (1978). A method for obtaining digital signatures and public-key cryptosystems. *ACM Communications*.

Rochwerger, B., Breitgand, D., Levy, E., Galis, A., Nagin, K., Llorente, I., Montero, R., Wolfsthal, Y., Elmroth, E., Caceres, J., Ben-Yehuda, M., Emmerich, W., and Galan, F. (2009). The reservoir model and architecture for open federated cloud computing. *IBM Journal of Research and Development*, 53(4):1–11.

Sakai, R., Ohgishi, K., and Kasahara., M., editors (2000). *Cryptosystems based on pairing*.

Sarbazi-Azad, H. and Zomaya, A. (2014). *A Cloud Broker Architecture for Multicloud Environments*, pages 760–768. Wiley-IEEE Press.

Sarigiannidis, P., Karapistoli, E., and Economides, A. A. (2017). Modeling the internet of things under attack: A g-network approach. *IEEE Internet of Things Journal*, 4(6):1964–1977.

Schneier, B. (1993). *Applied Cryptography: Protocols, Algorithms, and Source Code in C*. John Wiley & Sons, Inc., New York, NY, USA.

Sehgal, V. K., Patrick, A., Soni, A., and Rajput, L. (2015). Smart human security framework using internet of things, cloud and fog computing. In Buyya, R. and Thampi, S. M., editors, *Intelligent Distributed Computing*, pages 251–263, Cham. Springer International Publishing.

Shabut, A. M., Dahal, K. P., Bista, S. K., and Awan, I. U. (2015). Recommendation based trust model with an effective defence scheme for manets. *IEEE Transactions on Mobile Computing*, 14(10):2101–2114.

Shota Yamada, Nuttapong Attrapadungy, G. H. and Kunihiro, N. (2014). A framework and compact constructions for non-monotonic attribute-based encryption. In editor, editor, *Public-Key Cryptography - PKC 2014*.

Shropshire, J. (2014). Extending the cloud with fog: Security challenges and opportunities. In *AMCIS*.

Sicari, S., Rizzardi, A., Grieco, L., and Coen-Porisini, A. (2015). Security, privacy and trust in internet of things: The road ahead. *Computer Networks*, 76:146 – 164.

Sicari, S., Rizzardi, A., Miorandi, D., and Coen-Porisini, A. (2017). Dynamic policies in internet of things: Enforcement and synchronization. *IEEE Internet of Things Journal*, 4(6):2228–2238.

Singh, J., Pasquier, T., Bacon, J., Ko, H., and Eyers, D. (2016). Twenty security considerations for cloud-supported internet of things. *IEEE Internet of Things Journal*, 3(3):269–284.

Soleymani, S. A., Abdullah, A. H., Zareei, M., Anisi, M. H., Vargas-Rosales, C., Khuram Khan, M., and Goudarzi, S. (2017). A secure trust model based on fuzzy logic in vehicular ad hoc networks with fog computing. *IEEE Access*, 5:15619–15629.

Somu, N., M.R., G. R., Kirthivasan, K., and V.S., S. S. (2018). A trust centric optimal service ranking approach for cloud service selection. *Future Generation Computer Systems*, 86:234–252.

Sreenivasa Rao, Y. and Dutta, R. (2015). Fully secure bandwidth-efficient anonymous ciphertext-policy attribute-based encryption. *Security and Communication Networks*.

Tian, L.-q., Lin, C., and Ni, Y. (2010). Evaluation of user behaviour trust in cloud computing. In editor, editor, *International Conference of Computing and Applied System Modelling*.

Tian, W., Li, Y., Chen, Y., Tian, H., Cai, Y., Jia, W., and Wang, B. (2017). Fog-based evaluation approach for trustworthy communication in sensor-cloud system. *IEEE Communication Letters*, 21(11):2532–2535.

Vaquero, L. M. and Rodero-Merino, L. (2014a). Finding your way in the fog: Towards a comprehensive definition of fog computing. *ACM SIGCOMM Computer Communication Review*, 44(5).

Vaquero, L. M. and Rodero-Merino, L. (2014b). Finding your way in the fog: Towards a comprehensive definition of fog computing. *SIGCOMM Comput. Commun. Rev.*, 44(5):27–32.

Wang, E. K., Li, Y., Ye, Y., Yiu, S. M., and Hui, L. C. K. (2018a). A dynamic trust framework for opportunistic mobile social networks. *IEEE Transactions on Network and Service Management*, 15(1):319–329.

Wang, T., Zhang, G., Bhuiyan, M. Z. A., Liu, A., Jia, W., and Xie, M. (2018b). A novel trust mechanism based on fog computing in sensor–cloud system. *Future Generation Computer Systems*, pages 15619–15629.

Wang, Y. (2018a). Trust quantification for networked cyber-physical systems. *IEEE Internet of Things Journal*, 5(3):2055– 2070.

Wang, Y., Lu, Y.-C., Chen, I.-R., Cho, J.-H., Swami, A., and Lu, C.-T. (2014). Logittrust: A logit regression-based trust model for mobile ad hoc networks. In *6th ASE International Conference on Privacy, Security, Risk and Trust*.

Wang, Z. (2018b). Leakage resilient id-based proxy re-encryption scheme for access control in fog computing. *Future Generation Computer Systems*, 87:679 – 685.

Wen, Z., Yang, R., Garraghan, P., Lin, T., Xu, J., and Rovatsos, M. (2017). Fog orchestration for internet of things services. *IEEE Computer Society*, pages 16 – 24.

Wu, H. and Wang, W. (2018). A game theory based collaborative security detection method for internet of things systems. *IEEE Transactions On Information Forensics And Security*, 13(6).

Xia, H., Wang, G.-d., and Pan, Z.-k. (2016). Node trust prediction framework in mobile ad hoc networks. In *IEEE TrustCom/BigDataSE/ISPA*.

Xiaoyong, L., Huadong, M., Feng, Z., and Wenbin, Y. (2015). T-broker: A trust-aware service brokering scheme for multiple cloud collaborative services. *IEEE Transactions on Information Forensics and Security*, 10(7):1402–1415.

Xiaoyong, L. and Yuehua, Y. (2011). Trusted data acquisition mechanisms for cloud resource scheduling based on distributed agents. *China Communications*, 8(6):108–116.

Yan, Z., Zhang, P., and V. Vasilakos, A. (2014). A survey on trust management for internet of things. *Journal of Network and Computer Applications*, 42:120–134.

Yaoa, X., Chena, Z., and Tianb, Y. (2015a). A lightweight attribute-based encryption scheme for the internet of things. *Future Generation Computer Systems*.

Yaoa, X., Chena, Z., and Tianb, Y. (2015b). A lightweight attribute-based encryption scheme for the internet of things. *Future Generation Computer Systems*, 49:104–112.

Yassine, A., Nazari Shirehjini, A. A., and Shirmohammadi, S. (2015). Smart meters big data: Game theoretic model for fair data sharing in deregulated smart grids. *IEEE Access*, 3:2743–2754.

Yeh, L.-Y., Chiang, P.-Y., Tsai, Y.-L., and Huang, J.-L. (2018). Cloud-based fine-grained health information access control framework for lightweight iot devices with dynamic auditing and attribute revocation. *IEEE Transactions on Cloud Computing*, 6(2).

Yin, D., Zhang, L., and Yang, K. (2018). Blockchain-based mobile edge computing framework for secure therapy applications. *IEEE Access*, 6:72469–72478.

Zhou, Z., Huang, D., and Wang, Z. (2015). Efficient privacy-preserving ciphertext-policy attribute based-encryption and broadcast encryption. *IEEE Transactions on Computers*.