11-23-2023 10:30 AM

3Tier – AKA: A novel authentication using 5G communication for edge users in Cloud-Fog-Edge computing

Jiayi Zhang,

Supervisor: Ouda Abdelkader H., *The University of Western Ontario*
Co-Supervisor: Abu-Rukba Ra'afat, *American University of Sharjah*
A thesis submitted in partial fulfillment of the requirements for the Master of Engineering Science degree in Electrical and Computer Engineering

Abstract

The concept of fog computing architecture represents an extension of cloud computing, and it has gained significant traction across various domains like self-driving vehicle networks, smart cities, and smart homes. One of the key challenges of traditional cloud computing lies in the considerable distance between cloud data centers and the devices at the network's edge. This geographical gap results in substantial delays when processing data. To counteract this issue, fog computing deploys intermediate servers closer to the edge devices. This approach offers enhanced service efficiency and cost-effectiveness compared to conventional cloud computing.

However, despite its conceptual roots in cloud computing, fog computing introduces its own security challenges that cannot be fully addressed using solutions designed solely for the cloud environment. The primary concern revolves around ensuring security and privacy within fog computing networks, particularly in aspects related to authentication and key agreement. These challenges emerge from the distributed and dynamic nature of fog computing, which demands tailored security solutions.

This work proposes a novel mutual authentication and key agreement protocol specifically designed to address the security requirements of fog computing within the context of the edge-fog-cloud three-tier architecture, augmented by the integration of the 5G network.

The essence of the proposed protocol lies in leveraging the unique capabilities of the 5G network. By doing so, the protocol establishes secure communication channels across the different tiers of the architecture (edge, fog, and cloud). This secure channel establishment ensures dependable data transmission and offers protection against potential security threats, given the dynamic and diverse nature of the fog-based environment. The main objective of this protocol is to tackle the security concerns inherent in fog computing. It achieves this by incorporating robust and efficient mutual authentication and key management mechanisms. These mechanisms enhance the security within fog-based environments, where conventional security approaches might fall short.

This study enhances security in the cloud-fog-edge environment. The mutual authentication mechanism introduced in this thesis lays a foundation for seamless and secure communication among various entities in the distributed architecture. Capitalizing on 5G benefits, it advances secure communication for emerging cloud-fog-edge applications. A comparative analysis was undertaken, aligning the proposed protocol with established alternatives like TLS 1.3, 5G-AKA, and diverse handover protocols. Notably, our protocol boasts a mere 1280 bits for the complete communication costs in the authentication phase, accounting for nearly 30% less than other protocols. Furthermore, our handover protocol incurs only 2 signaling costs. The handover authentication computational cost for the edge user is notably low at 0.243 ms, amounting to just 10% of the computation costs of other protocols.

Keywords

Fog computing, Cloud computing, Mutual Authentication and Key agreement, 5G network, Handover, TLS 1.3, 4G LTE, Security and Privacy, Internet of Things (IoT).

Summary for Lay Audience

In the fast-paced world of modern communication, 5G technology has emerged as a powerful force, enabling faster and more reliable connections for devices. But how can we ensure that these connections are secure and trustworthy? That's where authentication protocols come into play.

Authentication protocols in 5G communication technology use advanced techniques like AES encryption and decryption to safeguard data and identity. AES encryption is like a secret code that only devices and the network can understand, making it incredibly difficult for unauthorized individuals to intercept or tamper with information.

But what happens when a user moves from one location to another while using their device? This is where handover authentication comes into play. Handover authentication ensures that as a user switches from one area to another, the connection between their device and the new fog node (a critical part of the network) remains secure and seamless.

Imagine someone is driving and streaming a movie on their phone. As they move from one city to another, the handover authentication ensures that their video keeps playing smoothly, without any interruptions due to security checks. It's like a seamless handoff of responsibilities from one security guard to another, ensuring the safety of data at all times.

This authentication process doesn't happen in isolation; it takes place in a three-tier environment, combining cloud, fog, and edge technology. The cloud is like a central hub that stores and processes vast amounts of data, while the fog represents smaller, distributed nodes closer to the user, making quick decisions and ensuring low-latency connections. The edge is the closest tier to the device, ensuring prompt responses for data requests.

By employing AES encryption and handover authentication in the cloud-fog-edge three-tier environment, 5G communication technology ensures that data remains secure, and connections stay strong, even as users move. This way, users can enjoy the benefits of fast and reliable communication without worrying about the safety of their information.

Acknowledgments

I am filled with gratitude as I reflect on the incredible journey of my master's program in Software Engineering at Western University. Completing this academic milestone would not have been possible without the invaluable support and encouragement from numerous individuals who have played pivotal roles in shaping my academic and personal growth.

First and foremost, I extend my deepest appreciation to my supervisors, Dr. Abdelkader Hassan Ahmed Ouda and Dr. Raafat Aburukba. Their unwavering dedication, expert guidance, and insightful feedback have been instrumental in shaping my research and pushing me to excel beyond my perceived limits. Their mentor-ship has not only enriched my technical knowledge but also instilled in me a passion for pushing the boundaries of software engineering. I am truly fortunate to have had the opportunity to work under their mentor-ship.

To my beloved girlfriend, I extend heartfelt thanks for her unconditional love, understanding, and patience throughout this journey. Her constant encouragement and belief in me have been a source of motivation during challenging times. She has been a steadfast companion, offering a listening ear and words of encouragement when I needed them the most. I am grateful for her presence in my life, as she has made every step of this journey more meaningful and enjoyable.

I also owe a debt of gratitude to my parents, whose unyielding support and belief in my abilities have been the cornerstone of my success. Their encouragement during moments of self-doubt has given me the strength to persevere and reach new heights. They have always been my pillars of strength, and I cherish the sacrifices they made to ensure I had the best opportunities in my academic pursuits.

To all my friends and colleagues at Western University who have shared the highs and lows of this master's program with me, I am grateful for the camaraderie and support we provided to one another. Our collaborative efforts have enriched the learning experience, making it memorable and rewarding.

Lastly, I would like to extend my appreciation to the staff and faculty at UWO, whose dedication and commitment to education have provided a nurturing environment for academic growth.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

3Tier – AKA	Three-Tier Architecture - Authenticated Key Agreement
4G LTE	4G Long Term Evolution
5G	Fifth generation
AES	Advanced Encryption Standard
AKA	Authentication and Key Agreement
AR	Augmented reality
ARPF	Authentication Credential Repository and Processing Function
AV	Authentication Vector
AVISPA	Automated Validation of Internet Security Protocols and Applications
BAN	Burrows-Abadi-Needham
BSs	Base stations
CN	Core Network
CPU	Central Processing Units
CS	Cloud server
DIDV	Dynamic login identity
DOS	Denial of service
EAP	Extensible Authentication Protocol
ECC	Elliptic curve cryptography
ECIES	Elliptic Curve Integrated Encryption Scheme

eMBB	Enhanced mobile broadband
EADA	Edge-assisted decentralized authentication
ENs	Edge nodes
EOP	Elevation of privilege
EPS – AKA	Evolved Packet System - Authentication and Key Agreement
eSIM	Embedded Silence iz Mine
FA	Foreign agent
FNs	Fog nodes
GPS	Global Positioning System
HA	Home agent
HKDF	Hash-based message authentication code key-derivation function
IaaS	Infrastructure-as-a-Service
ID	Identity
IIoT	Industrial Internet of Things
IoT	Internet of Things
IoV	Internet of Vehicles
LA	Local authority
LEOs	Low-earth-orbit satellites
MEC	Multi-Access Edge Computing
MD	Mobile device

MITM	Man-In-The-Middle
mMTC	Massive machine-type communications
NCC	Network Control Center
NFV	Network Functions Virtualization
NTRU	Number Theory Research Unit
OS	Operating systems
PaaS	Platform-as-a-Service
PKI	Public-key infrastructure
RA	Registration authority
RAN	Radio Access Network
ROR	Real-Or-Random
RRT	Round Trip Time
RS	Registration server
RSUs	Road-side units
SaaS	Software-as-a-Service
SDN	Software-Defined Networking
SIM	Silence iz Mine
SLAs	Service level agreements
SOA	Service Oriented Architecture
SPAN	Security Protocol Animator

SUPI	Subscription Concealed Identifier
TA	Trusted authority
TLS	Transport Layer Security
UDM	Unified Data Management
UE	User Equipment
URLLC	Ultra-reliable and low-latency communications
VANETs	Vehicular ad hoc networks
VFSs	Vehicular fog services
VR	Virtual reality

Chapter 1

1 Introduction

User security refers to the measures, practices, and technologies implemented to protect users' personal information, digital identity, and online activities in various digital environments, such as websites, applications, online services, and communication platforms [1]. User security ensures that individuals can interact with digital platforms safely and confidently without the risk of unauthorized access, data breaches, identity theft, or other cyber threats. User security is crucial not only for protecting individuals but also for building trust in digital platforms and services [2]. Organizations must prioritize user security as part of their broader cybersecurity strategy, consistently updating their security measures to adapt to evolving threats and ensure a safe and secure digital experience for users. Security in a cloud-fog-edge computing environment refers to the set of measures, practices, and technologies to protect data, resources, applications, and communication within the context of these distributed computing paradigms. One of the key aspects of security in a cloud-fog-edge computing environment is authentication. Establishing robust authentication processes to confirm entities' identity and determine their access rights based on authorization policies [3].

The cloud-fog-edge (3-tier) computing environment is a distributed computing architecture that combines the capabilities of cloud computing, fog computing, and edge computing to provide a comprehensive and flexible framework for processing, storing, and managing data and services. This environment takes advantage of the strengths of each computing paradigm to address a wide range of applications and scenarios. Here's an overview of each component within the cloud-fog-edge computing environment:

- Cloud Computing: The cloud component provides a centralized, scalable, and often remote infrastructure for storing and processing data and running applications. Cloud resources offer extensive computational power and storage capacity, making them suitable for tasks requiring significant resources and complex data analysis [4].

- Fog Computing: Fog computing extends cloud capabilities to the network edge, closer to the data source and end-users. Fog nodes, which can include routers, switches, and gateways, process and filter data locally, reducing latency and supporting real-time applications. Fog computing is particularly useful for scenarios where low-latency interactions and immediate responses are critical, such as IoT applications [5].
- Edge Computing: Edge computing brings computation even closer to the data source, often directly on the devices themselves. This minimizes the need for data transmission to distant data centers, reducing latency and bandwidth usage. Edge computing is ideal for applications demanding rapid data processing and quick decision-making, such as autonomous vehicles and robotics [6].

The cloud-fog-edge computing environment envisions a holistic ecosystem that efficiently leverages resources across the entire spectrum.

User authentication involves verifying an identity asserted by or on behalf of a system entity[7], [8]. In the cloud-fog-edge environment, authentication plays a crucial role in ensuring secure communication, protecting sensitive data, and mitigating the risks associated with unauthorized access or malicious activities [9]. This environment encompasses a wide range of entities, including cloud servers, fog nodes, edge devices, and users, each with its own unique characteristics and requirements. The distributed nature of the cloud-fog-edge environment poses challenges to authentication. Due to the mobility of edge users, authentication protocol requires a function of fast handover.

Cloud computing, at its core, refers to the delivery of on-demand computing resources and services over the internet. It provides users with a flexible and scalable infrastructure, enabling them to access computational power, storage, and software applications as per their requirements, without the need for extensive on-site hardware and software installations [4]. The cloud computing model has gained immense popularity due to its ability to lower costs, enhance collaboration, and support a wide range of applications and services [10]. Some challenges in cloud computing are as follows:

- Security and Privacy Concerns: Security is a concern in cloud computing due to the sharing of resources and sensitive data across multiple users. Issues such as data breaches, unauthorized access, and inadequate control over data are prevalent [11].
- Data Loss and Recovery: As data is stored remotely, there is a risk of data loss due to hardware failures, software bugs, or human errors. Ensuring reliable data backup and recovery mechanisms is crucial [12].
- Performance and Latency: Cloud computing relies on the internet for access, which can introduce latency and impact application performance, especially for applications requiring real-time processing. Moving large volumes of data to and from the cloud can be slow and bandwidth-intensive, especially for organizations with limited network capacity [13].

To address the latency issue in cloud computing for latency-sensitive applications, fog computing was proposed in 2012 by Cisco [5]. Fog computing, a paradigm that extends cloud computing capabilities to the edge of the network, has emerged as a promising solution to address the limitations of centralized cloud computing in the era of Internet of Things (IoT) and edge computing [14]. Fog computing complements cloud computing by bringing computation, storage, and networking resources closer to the edge devices [15], [16]. By leveraging nearby fog nodes located at the network edge, fog computing reduces latency and improves real-time data processing. This distributed computing model empowers edge devices to perform local computation and offload selective tasks to fog nodes or the cloud, optimizing the overall system performance [17].

The convergence of fog, cloud, and edge computing brings significant benefits in data processing, storage, and decision-making. However, for this convergence to be effective, a robust and high-speed communication infrastructure is required. The 5G network, with its ultra-fast data transfer rates, low latency, and high capacity, provides the necessary capabilities to meet the demanding communication requirements of fog, cloud, and edge computing environments [18]. One of the key advantages of 5G network connectivity is its ability to support massive machine-type communications (mMTC) and ultra-reliable and low-latency communications (URLLC). mMTC enables the efficient handling of a

large number of IoT devices, sensors, and actuators, which are characteristic of edge and fog computing environments [19]. These devices can generate vast amounts of data that need to be transmitted and processed in real-time. Furthermore, the 5G network's mMTC capabilities ensure reliable and efficient communication between these devices and fog/cloud resources.

One of the defining characteristics of the cloud-fog-edge environment is its hierarchical nature. At the cloud level, centralized data centers and servers provide immense processing power and storage capacity, catering to applications with significant computational and storage requirements [4]. The fog layer, located closer to the network edge, comprises intermediate nodes and devices that offload computational tasks from the cloud and support real-time analytics [5]. Finally, the edge layer consists of devices and sensors at the network edge, enabling localized computation and immediate response [6].

The distributed nature of the cloud-fog-edge environment offers several advantages. It reduces the latency in data processing and enables real-time decision-making, making it ideal for applications that require quick response times and low latency, such as autonomous vehicles, augmented reality, and industrial automation. Additionally, by bringing computation and storage closer to the edge, the cloud-fog-edge environment reduces network congestion and conserves bandwidth, as only selective data needs to be transmitted to centralized resources.

In the realm of cloud-fog-edge environment, various components exist, including edge devices such as mobile phones, laptops, and smart city infrastructure, as well as fog nodes and cloud data centers. The role of the fog node is to process tasks created by the edge and transmit data or less critical tasks to the cloud services. This computing environment exhibits several notable characteristics as follows [5], [20], [21]:

- Compared to a cloud data center, a fog node is situated closer to the end user, resulting in lower latency. This reduced latency translates into higher operational efficiency.
- Fog computing empowers endpoints to provide premium services at the network's edge, and it also enables location awareness. Within cloud-fog-edge environment,

location technologies like Global Positioning System (GPS) can be utilized to determine the position of devices. Consequently, cloud-fog-edge environments, such as the Internet of Vehicles (IoV) based on fog computing, can leverage location awareness.

- Mobility refers to the capability of seamlessly accessing and utilizing computing resources, data, and services across a range of devices and locations. It allows users and applications to maintain consistent experiences and interactions regardless of their physical location or device type. Mobility ensures that data and services are available to users whenever and wherever they need them, enhancing the flexibility and convenience of computing environments.
- Scalability, which refers to the ability to adjust the number of IoT devices to meet changing demands, is a distinguishing feature of cloud-fog-edge environment. Cloud-fog-edge environment excels in terms of scalability.
- Geographical distribution is another key characteristic of cloud-fog-edge computing paradigm. Similar to the centralized cloud, fog computing can offer services over a wider geographic range. This characteristic aligns with the concept of scalability.
- Interoperability is crucial, particularly in the context of IoT devices, which often involve collaboration between different providers. Therefore, the components of the fog infrastructure must possess the ability to interoperate. For example, various sensors are deployed across the city to monitor traffic conditions. These sensors are manufactured by different companies and may use different communication protocols to transmit data. Interoperability ensures that traffic sensors can communicate with various fog nodes, regardless of the sensor's manufacturer or communication protocol.

1.1 Research Motivation

In cloud computing, performance limitations arise from the reliance on network connectivity and the distance between users and the cloud data centers. Latency and network congestion can impact the response time of cloud-based applications, especially for real-time or latency-sensitive tasks [22]. Additionally, the availability and reliability

of cloud services depend on the network infrastructure and the service level agreements (SLAs) provided by cloud service providers[23]. Unpredictable service downtime or performance fluctuations can disrupt business operations and undermine user satisfaction. Security concerns represent another critical limitation of cloud computing. While cloud service providers invest heavily in security measures, storing sensitive data and applications on external servers raises concerns about data privacy, integrity, and confidentiality. Furthermore, the risk of data breaches and unauthorized access to cloud resources remains a significant concern that requires robust security mechanisms and protocols [24].

Fog computing also brings forth security and authentication challenges that must be carefully addressed [20]. Security and authentication challenges in fog computing arise due to the distributed and heterogeneous nature of the environment, where a variety of devices, communication protocols, and data processing locations are involved. These challenges must be addressed to ensure the confidentiality, integrity, and availability of data and services [25]. Addressing these challenges requires a comprehensive security strategy that combines encryption, authentication, access controls, and monitoring mechanisms. It also involves the collaboration of fog computing stakeholders to develop standardized security protocols and best practices that suit the unique characteristics of the fog computing paradigm [26].

Authentication, is crucial to ensure that only authorized entities have access to fog-based resources. In fog computing, the distributed nature of the infrastructure introduces new authentication challenges. Edge devices, fog nodes, and cloud servers need to authenticate each other securely to prevent unauthorized access and data manipulation. However, the resource-constrained nature of edge devices, the dynamic nature of fog computing environments, and the latency requirements pose unique challenges in designing lightweight and efficient authentication mechanisms that can adapt to the heterogeneous and transient nature of the network [27].

The 5G network plays a crucial role in enabling efficient and reliable communication between fog, cloud, and edge computing environments. However, while the 5G network

brings numerous benefits to fog, cloud, and edge computing, there are also challenges that need to be addressed. These challenges include network security, scalability, and efficient resource allocation [18]. Ensuring the security and privacy of data transmitted over the 5G network is crucial to prevent unauthorized access and protect sensitive information.

The cloud-fog-edge environment also presents challenges that need to be addressed. Efficient resource management, workload distribution, and task offloading across different layers of the environment require intelligent algorithms and dynamic adaptation mechanisms [28]. Additionally, ensuring security, privacy, and data integrity becomes critical as data is distributed across multiple tiers and devices in the environment. The heterogeneity of devices, varying computational capabilities, and resource constraints further complicate the development of robust and scalable solutions [29].

The existing authentication protocols designed for cloud computing are not suitable for fog computing [30]. This is primarily due to the heterogeneity and interoperability challenges associated with fog computing, which introduce significant security concerns in terms of identity authentication. Fog nodes interact with multiple edge users simultaneously [31]. Therefore, it is imperative to prioritize research and development of lightweight authentication approaches in order to facilitate low-latency services [32].

In the context of Fog Computing, where IoT edge users are served, the collection of sensitive user data poses a potential risk to user security. Fog nodes possess the capability to detect the identity and location of IoT edge users, as well as monitor their behaviors. Consequently, if an attacker successfully identifies a user, compromising the security of the IoT edge user becomes relatively easy [33]. These attacks can originate from an adversarial internal node, a denial of service (DOS) attack, or employ a Man-In-The-Middle (MITM) attack model. As a result, the fog computing environment may become vulnerable to threats such as information disclosure, data damage, resource manipulation, and similar security breaches [34].

1.2 Research Objective

The work aims to design a lightweight mutual authentication protocol tailored for the edge-fog-cloud environment, specifically integrated with the 5G network. This protocol is to provide a secure and efficient authentication mechanisms that address the unique challenges posed by the distributed nature of the cloud, fog, and edge computing paradigms. This involves designing mechanisms that allow for the seamless transfer of authentication credentials and cryptographic keys, ensuring continuous service without compromising security while accommodating the dynamic nature of the 5G network and the unique characteristics of edge-fog-cloud computing. By incorporating the lightweight cryptographic technique with Advanced Encryption Standard (AES) [35], the goal is to enhance authentication's overall security and performance in this complex computing environment.

To achieve this objective, this work analyzes the unique security requirements and constraints of the cloud-fog-edge environment, considering factors such as resource limitations, network latency, and potential attacks during first authentication and handover scenarios. Based on this analysis, lightweight cryptographic techniques and protocols are designed and developed to enable secure and efficient authentication.

Furthermore, this work explores the integration of handover authentication functions into the designed protocol, enabling smooth transitions and minimizing authentication delays or disruptions.

The designed protocols are extensively evaluated through simulations to assess its security, performance, and scalability in cloud-fog-edge scenarios. By achieving these objectives, the research endeavors enhance the security in the context of emerging 5G-enabled applications and services that heavily rely on edge and fog computing infrastructures. The research outcomes contribute to advancing the authentication protocols in distributed computing environments, providing a lightweight solution with handover authentication capabilities that ensure secure and seamless user experiences in the cloud-fog-edge paradigm.

1.3 Research Methodology

In order to address the challenges of authentication in the cloud-fog-edge environment, the research methodology combines theoretical exploration, protocol design, and performance evaluation. The following outlines the key steps involved in the research methodology:

Literature Review:

Conduct an extensive literature review to comprehensively understand the existing authentication and privacy preservation techniques in the cloud-fog-edge environment. This review will help identify the state-of-the-art approaches, their limitations, and the gaps in the current research.

Problem Formulation:

Define the research problems and objectives related to authentication in the cloud-fog-edge environment. Clearly articulate the research questions and hypotheses that will guide the study.

Theoretical Exploration:

Investigate theoretical concepts and foundations of authentication protocols, cryptographic algorithms, and secure communication protocols. Analyze the strengths and weaknesses of existing methods and explore potential improvements and innovations in the context of the cloud-fog-edge environment.

Protocol Design and Development:

Based on theoretical exploration, design novel authentication protocols specifically tailored for the cloud-fog-edge environment. Develop algorithms and techniques that address the challenges of heterogeneity, resource constraints, and the dynamic nature of the environment.

Performance Evaluation and Comparative Analysis:

Perform a comparative analysis of the proposed solutions against existing authentication and privacy preservation techniques in terms of performance, security, preservation, and usability. Identify the strengths and weaknesses of the proposed approaches and highlight their advantages over the state-of-the-art methods.

Conclusion:
In the conclusion chapter, summarize the key findings, contributions, and implications of the research. Reflect on the limitations of the proposed solutions and provide suggestions for future research directions.

1.4 Research Contribution

This work introduces a novel and lightweight mutual authentication and key agreement protocol called 3Tier – AKA (Three-Tier Architecture - Authenticated Key Agreement). The proposed protocol aims to achieve mutual authentication and session key agreement in the three-tier architecture. Notably, the protocol prioritizes the protection of user privacy by leveraging anonymous user identities and harnessing the communication efficiency offered by the 5G network.

In our designed 3Tier – AKA protocol, a distinctive feature is that the personal information and confidential session keys associated with edge users and other entities are deliberately not retained within the domains of the 5G server providers. This strategic approach significantly bolsters the preservation of security within the protocol's framework.

Moreover, when juxtaposed with analogous protocols that address similar concerns, our inventive solution exhibits a notable reduction in computational complexity, storage requirements, and signaling overhead. This optimization strategy leads to a demonstrably heightened level of efficiency and resource utilization within the protocol's operational ecosystem. This innovative enhancement not only enhances the overall performance of the protocol but also aligns with contemporary demands for streamlined and resource-efficient solutions in the realm of edge and 5G networking. Moreover, the protocol leverages AES encryption/decryption operations to cater to the computing limitations

typically encountered by edge users. This adaptation ensures that the protocol remains suitable for deployment in resource-constrained edge devices while maintaining a prominent level of security.

This work contributes to the field by providing an effective and lightweight solution for mutual authentication and session key agreement in three-tier architectures. The protocol's emphasis on user security, reduced computational and storage requirements.

1.5 Research Outline

The remaining chapters of the paper are organized as follows: Chapter 2 briefly summarizes the literature review on fog computing environment and existing TLS 1.3 limitations. Chapter 3 presents the characteristics of the environment and chapter 4 details the proposed 3Tier – AKA protocol in detail. Chapter 5 describes security and feature analysis. Performance evaluation and comparison with other related schemes are performed in Chapter 6. Finally, Chapter 7 concludes the paper.

Chapter 2

2 Literature Review and Background

This chapter provides an overview of the current protocols and background information pertaining to the environment.

2.1 Background

2.1.1 Fog Computing

Fog computing is an emerging paradigm that extends cloud computing capabilities to the network edge, closer to end-users and devices [22]. It leverages the concept of the 3-tier architecture, which consists of three distinct layers: the cloud, the fog, and the edge[36], [37], [27], [38]:

The cloud represents the centralized data centers with high computational and storage capabilities. The cloud tier constitutes a collection of robust servers housed within the physical infrastructure, designed to efficiently manage, store, and process vast volumes of data. This tier offers seamless integration with the edge tier. Nevertheless, a significant challenge arises due to its limitation in meeting the low latency demands of certain applications. The cloud boasts substantial processing and caching capacities, enabling users and applications to access its resources from anywhere, at any given moment.

Fog computing is characterized as a dynamic environment where an extensive array of diverse wireless and occasionally self-governing devices collaboratively communicate with each other and the network. This collaboration enables them to execute storage and processing tasks independently, without reliance on external entities. These tasks can cater to fundamental network functions or facilitate the deployment of novel services and applications within a sandboxed environment. The fog encompasses intermediary devices such as routers, gateways, and switches. Fog nodes possess both computational power and storage capabilities, facilitating seamless interaction and resource-sharing among them. These fog compute nodes are interconnected with the cloud architecture and periodically transmit both raw and processed data to the cloud for further utilization.

The edge layer comprises the devices at the network periphery, including sensors, smartphones, smart vehicles, and IoT devices. Edge nodes exhibit constrained compute, storage, and networking capabilities. In the edge tier of the Internet of Things (IoT) architecture, IoT devices play a vital role in sensing diverse events, executing constrained tasks, and relaying the unprocessed data they collect to the fog tier. These edge nodes typically comprise sensors, actuators, modest computing elements, networking infrastructure, and embedded operating systems. They are equipped with software components and custom-developed programs tailored to their specific functions.

This hierarchical structure allows for distributed computing, enabling faster response times, reduced network congestion, and improved scalability in dynamic and latency-sensitive applications.

While fog computing offers several advantages, it also introduces unique security challenges and vulnerabilities. Some common vulnerabilities in fog computing include unauthorized access, data breaches, denial of service (DoS) attacks, compromised intermediary devices, and insecure communication channels [39]. The dynamic nature of fog computing environments, along with the heterogeneity and large-scale deployment of devices, makes them susceptible to various security threats [20]: Data Privacy and Confidentiality: Fog computing involves processing and storing data at the edge devices, which increases the risk of unauthorized access and data breaches. Since these devices may not have robust security mechanisms, sensitive data could be exposed to potential attackers [40]. Resource Constraints: Many edge devices in fog computing environments have limited resources, including processing power, memory, and energy. This limitation makes it challenging to implement strong security measures, leaving them more vulnerable to attacks [20]. Interoperability Issues: Heterogeneity of devices and platforms in fog computing can lead to interoperability challenges, which attackers might exploit to gain unauthorized access or disrupt services [22]. Data Integrity: Fog computing involves data processing and storage at the edge, which opens up opportunities for data tampering and integrity breaches. Ensuring the integrity of data in such a decentralized environment becomes a significant concern [30]. These vulnerabilities necessitate the development of

robust authentication mechanisms and privacy-preserving techniques to mitigate potential risks and safeguard sensitive data in fog computing environments.

2.1.2 Authentication Protocols in Distributed Systems

Authentication plays a critical role in ensuring secure communication and access control in distributed systems [41]. In fog computing, the distribution environment refers to the physical infrastructure and network layout where fog nodes are strategically distributed. Fog nodes are geographically dispersed computing devices that are placed closer to the edge of the network, closer to the data sources and end-users [17]. It aims to bring computational resources, storage, and services closer to the edge devices and users, reducing latency and improving the overall performance of the system. Authentication protocols are used to verify the identities of entities, such as fog nodes, edge devices, and cloud services, within a distributed computing environment [42]. These protocols ensure secure communication and establish trust between the entities involved. The primary goal of authentication protocols in fog computing is to prevent unauthorized access, protect data integrity, and ensure the authenticity of the participants [40]. Various authentication approaches have been proposed in the literature to authenticate users, devices, and services within distributed environments. These approaches include password-based authentication, public-key infrastructure (PKI), certificate-based authentication, challenge-response protocols, and token-based authentication [43]. Each method has its own advantages and limitations, and the selection of an appropriate authentication mechanism depends on the specific requirements and characteristics of the system.

2.1.3 5G Communication Network Authentication

The 5G (fifth-generation) architecture is designed to provide higher data rates, lower latency, increased capacity, and improved user experience compared to its predecessor, 4G/LTE. It is based on a flexible and virtualized network infrastructure, enabling the efficient delivery of a wide range of services, including enhanced mobile broadband (eMBB), massive machine-type communications (mMTC), and ultra-reliable low-latency communications (URLLC) [44]. eSIM, short for embedded Silence iz Mine (SIM), is a technology that enables the use of a digital SIM card directly embedded within a device,

such as a smartphone, tablet, or wearable, without the need for a physical SIM card [45]. It is designed to replace the traditional removable SIM cards used in mobile devices. The 5G architecture is characterized by the following key components [46], [47], [48], [49]:

a) User Equipment (UE): The mobile devices used by end-users, such as smartphones, tablets, and IoT devices.

b) Radio Access Network (RAN): The RAN is responsible for connecting UEs to the core network. It includes the base stations, antennas, and other radio equipment.

c) Core Network (CN): The core network handles data processing, authentication, and service delivery. It is a virtualized network that can be optimized for different use cases.

d) Network Slicing: 5G supports network slicing, allowing the creation of multiple virtual networks with specific performance characteristics to cater to different types of services.

e) SDN (Software-Defined Networking) and NFV (Network Functions Virtualization): 5G leverages SDN and NFV technologies to enable network flexibility, scalability, and cost efficiency.

In the context of 5G, fog computing is integrated into the network through MEC (Multi-Access Edge Computing). MEC nodes are deployed at the edge of the 5G network, in close proximity to the end-users [50]. These nodes can host applications, services, and virtualized network functions, enabling data processing and service delivery closer to the point of data generation. By reducing the distance data has to travel, fog computing helps lower latency and improve the overall efficiency of the 5G network [51]. Fog computing within the 5G network is particularly beneficial for latency-sensitive applications like augmented reality (AR), virtual reality (VR), autonomous vehicles, industrial automation, and real-time gaming [52].

The 5G network employs several authentication protocols to ensure secure communication between the user equipment (UE) and the network. Some of the key authentication protocols used in 5G are:

a) Authentication and Key Agreement (AKA) [53]: AKA is a challenge-response-based protocol used during the initial connection setup between the UE and the 5G core network. It ensures that only legitimate UEs are granted access to the network.

b) Extensible Authentication Protocol (EAP) [54]: EAP is an authentication framework that supports multiple methods for authentication. It allows various authentication mechanisms to be used based on the specific needs of the user and network.

c) Transport Layer Security (TLS) [55]: TLS is a cryptographic protocol used to secure data transmission between the UE and network elements. It ensures privacy and integrity during communication.

d) Secure Authentication Vector (AV) [53]: This protocol is responsible for generating authentication vectors used by the AKA protocol for mutual authentication between the UE and the network.

e) Subscription Concealed Identifier (SUPI) [53]: The SUPI is used to protect the user's permanent identity by concealing it behind temporary identifiers during authentication procedures.

These authentication protocols work together to establish a secure connection and protect user privacy within the 5G network. They play a vital role in ensuring the integrity and security of the communication between the user and the network components.

2.2 Literature Review

As an extension of cloud computing, fog computing is closer to edge devices. However, the existing cloud computing authentication protocols cannot directly apply to fog computing [56]. In fog computing, many authentication protocols have been proposed [57], [58], [59], [60], [61].

Zhong et al. [62] address securing communication and privacy in fog-based vehicular ad hoc networks (VANETs). They propose an approach for secure and lightweight conditional privacy-preserving authentication. The paper aims to establish secure communication while preserving privacy in VANETs. Traditional authentication methods are inefficient and may expose sensitive information, such as vehicle ID, location data, movement path and scope of everyday routines. The work focused on designing a secure and efficient authentication mechanism. The proposed approach assumes fog based VANETs where vehicles and fog nodes communicate. The proposed mechanism employs conditional privacy, where vehicles disclose only necessary information for authentication while preserving anonymity. Cryptographic techniques like bilinear pairings and elliptic curve cryptography (ECC) are used for secure and efficient authentication, minimizing computation and communication overhead. They present an evaluation of the proposed authentication scheme through simulations. It demonstrates a balance between security and efficiency in VANETs. The conditional privacy mechanism establishes secure communication while minimizing computational and communication overhead. The approach is suitable for resource-constrained vehicular environments. However, scalability aspects and the impact of dynamic network conditions are not extensively addressed. Further research is needed to enhance the scalability, particularly in scenarios with many vehicles and fog nodes, and to investigate the effects of varying vehicular density and mobility patterns.

In [63], Rudri et al. proposed a mutual authentication based on Elliptic Curve Cryptography (ECC) and one-way hash functions. In this paper, they focus on the challenge of establishing mutual authentication between fog nodes and end-users in fog computing environments. The scheme aims to authenticate communication between the cloud, fog, and edge devices in a resource-constrained and dynamic network. The

proposed fog-based mutual authentication scheme uses low-cost primitives, specifically Elliptic Curve Cryptography (ECC) and one-way hash functions. By leveraging these lightweight cryptographic techniques, the proposed approach establishes secure mutual authentication while minimizing computational and communication overheads. The scheme undergoes a security analysis, demonstrating protection against known attacks. Validation using the Security Protocol ANimator (SPAN) of Automated Validation of Internet Security Protocols and Applications (AVISPA) tool confirms its effectiveness. Comparative evaluations showcase superiority over existing schemes. However, this protocol requires the edge user to store an extra identity (ID). Cloud server and fog server are not authenticated to each other.

The protocol used in [64] is a lightweight authentication protocol designed for resource-constrained devices in the Industrial Internet of Things (IIoT). The IIoT ecosystem faces numerous security vulnerabilities, including industrial espionage and sabotage. To mitigate these risks, robust authentication mechanisms are essential. However, resource-constrained devices, which are prevalent in the IIoT, pose challenges due to their limited computing capabilities. The proposed authentication protocol is designed to operate within the IIoT environment. This environment comprises sensors, networks, and services that connect and control production systems. It assumes the presence of resource-constrained devices, which may have limitations in terms of computational power, memory, and energy efficiency. The authentication protocol proposed in the paper uses lightweight operations, including xor, addition, and subtraction, along with a hash function. The protocol minimizes communication overhead by facilitating authentication with just four message exchanges between the participating entities. The proposed authentication protocol undergoes rigorous security assessment using formal methods, including the Automated Validation of Internet Security Protocols and Applications (AVISPA) tool and Burrows-Abadi-Needham (BAN) logic. Furthermore, a performance and security comparison with state-of-the-art protocols reveals its efficient performance for resource constrained IIoT devices. The protocol achieves higher security levels comparable to computationally expensive schemes.

The papers [65] [66] have raised mutual authentication in Internet of Vehicle (IoV) environment. Han et al. [65] has proposed an anonymous-authentication scheme based on fog computing for vehicular ad hoc networks (VANETs). The problem lies in the privacy concerns associated with using real identities for vehicle authentication in VANETs. Existing privacy-protection schemes rely on anonymous authentication, but face challenges related to network congestion and the process of updating anonymous information, resulting in poor real-time performance and potential key leakage. The proposed scheme operates within the VANET environment, assuming the presence of vehicles, road-side units (RSUs), and a trusted authority (TA). The proposed scheme operates within the VANET environment, assuming the presence of vehicles, road-side units (RSUs), and a trusted authority (TA). The focus is on improving the authentication process and reducing communication burdens between vehicles and RSUs. However, there is an authority called local authority (LA). This authority is responsible for management of vehicles anonymous information. Utilizing a third-party authority will lead to elevated expenses, as regular commissions need to be paid to these entities. Also, there is no authentication between roadside units. In [66], Chen et al. proposed a mutual authentication protocol for IoV environment. This protocol can tackle various security concerns, such as identity compromise, identity theft, and replay attacks. In this paper [66], there is a comprehensive security analysis of the proposed protocol, establishing its effectiveness in thwarting these specific attack vectors. Each vehicular user must register in trust authority (TA), then TA issues a smart card to them. The vehicular user can use the smart card to authenticate to roadside unit. However, it is significant to note that certain vulnerabilities, such as distributed denial-of-service attacks are not specifically addressed in their study. Once the attacker sends multiple authentication requests to TA, TA must compare dynamic login identity (DIDV) and value CV to the stored value. After this, TA creates a random integer β and a session key Ks and saves them to its storage. This will cause waste of storage and denial-of-service.

Blockchain and consensus mechanism are used for anonymous authentication in [67]. With the rapid advancements in modern vehicles and distributed fog services, the expansion of vehicular fog services (VFSs) is becoming increasingly important, necessitating their presence across multiple geographically dispersed datacenters.

Consequently, the need for cross-datacenter authentication arises. However, traditional cross-datacenter authentication models are ill-suited for the specific scenario of high-speed moving vehicles accessing VFSs. These models have either disregarded user privacy or failed to meet the time constraints associated with driving vehicles. In this protocol, only the fog node closest to the vehicle is required to authenticate vehicles, after which the fog node broadcasts the authentication result to other fog nodes and records the results to blockchain. Therefore, this paper did not consider fog node compromise attack. Once a fog node is controlled or compromised, the reported result to the blockchain could be misleading.

In [68], Ibrahim et al. proposed a mutual authentication scheme called Octopus for the Edge Fog Cloud network architecture, which utilizes a master secret key for new users to authenticate themselves to fog server. However, the scheme has a limitation: it openly transmits the user's identity over a public channel, compromising user anonymity. Also, Octopus is primarily designed for stationary smart cards and devices, which increases the risk of interference from masquerading servers. In the context of a large-scale fog computing environment, reusing the same master password can pose a significant security risk.

Cloud computing and IoT convergence have limitations for low-latency and mobile applications. Fog computing bridges this gap by bringing computation closer to end devices. However, remote and unprotected fog nodes require secure solutions, especially in IoT healthcare systems. In [69], Jia et al. proposed a fog driven IoT healthcare system authentication key agreement protocol. Fog nodes connect cloud data centers and end devices, aiming for efficient and secure healthcare services. It considers resource-constrained fog nodes, mobile healthcare devices, and low-latency communication requirements. The protocol leverages bilinear pairings to establish secure cryptographic keys among the entities involved, ensuring authentication and secure communication. Also, it employed an Authenticated Key Agreement (AKA) protocol, involving three components: fog node, cloud server, and sensors. It introduces a security model, provides formal security proof, and defends against common attacks. Performance evaluation considers communication and computation costs. Results demonstrate the protocol's

secure and efficient authentication and key exchange for fog-driven IoT healthcare systems. However, there are some limitations with this protocol. The attacker can perform a password guessing attack. Additionally, it costs a huge number of computational resources.

Mobile edge computing (MEC) addresses limitations in cloud computing but ensuring security in MEC settings is challenging. Network operators must consider security and privacy as critical challenges for establishing an MEC ecosystem. In [70], an identity-based anonymous authenticated key agreement protocol is proposed for MEC environments. This protocol achieves mutual authentication in a single message exchange round while ensuring user anonymity and un-traceability. This identity-based authentication scheme utilized elliptic curve cryptography and bilinear pairing techniques in a mutual authentication process. It aims to provide user anonymity and untrace ability. However, due to its characteristic, the attacker can attempt to break into user accounts by repeatedly trying different user and password combinations. Furthermore, the computational costs of this protocol are significantly high, primarily attributed to the pairing operations it involves.

Dewanta, et al. [71] propose a mutual handover authentication in vehicular network environment. The paper focuses on the challenge of secure fog computing service handover in vehicular networks. It aims to establish mutual authentication between vehicles and fog nodes (FNs) to ensure the integrity and privacy of vehicular network systems. The environment consists of vehicles, fog nodes deployed on roadside units (RSUs), and a cloud server (CS) that facilitates the authentication process. The scheme utilizes one-way hash functions and exclusive-or operations to ensure its lightweight nature. During the login and service request phases, the cloud server distributes credentials for on-the-road authentication between vehicles and fog nodes. During the handover process, vehicle and fog node can perform mutual authentication by using login credentials. The approach achieves computational efficiency by providing faster computation and reduces the total message size compared to previous authentication schemes in similar environments. This scheme's validation using the SPAN software based on AVISPA confirms its effectiveness in achieving mutual authentication goals

and its resilience against replay and man-in-the-middle attacks. However, this paper does not discuss how CS distributes user's credentials to the specific fog nodes.

The paper [57] introduces a novel anonymous handover authentication scheme for fog computing. The paper addresses the problem of secure handover authentication for mobile devices in fog computing environments. Handovers, where mobile devices transition between different fog access points, require authentication mechanisms that ensure secure communication while maintaining efficiency and preserving user privacy. In this protocol, edge users and fog nodes have to register in a registration authority (RA), then retrieve their unique identity and compute the corresponding pseudo-identity. After edge user login and authenticate with one fog node, they move to another fog node, the pre-negotiation between old and new fog node happens. Then edge user can authenticate with new fog node. The encryption methods used in this protocol are all lightweight, such as concatenation and bitwise XOR operation. The proposed FogHA scheme leverages symmetric trivariate polynomials to provide low-latency authentication while ensuring security, user anonymity, and resistance against known attacks. The authors analyze the security of FogHA using the Real-Or-Random (ROR) model. The analysis demonstrates the semantic security of the scheme, indicating its resilience against potential attacks. An informal security analysis displays FogHA's ability to resist various known attacks, including mutual authentication, replay attacks, man-in-the-middle attacks, impersonation attacks, and more.

Yang, et al. [61] proposed a threshold mutual authentication protocol which supports fast handover. The paper focuses on the problem of secure and efficient access authentication in vehicular networks. Existing authentication protocols often lack consideration for attacks like single points of failure and fail to effectively reduce authentication delays. The authors aim to tackle these challenges and provide a decentralized authentication architecture that enhances security and efficiency. The paper assumes a vehicular network environment comprising a registration server (RS), edge nodes (ENs) such as roadside units (RSUs) and base stations (BSs), and vehicles. The RS is considered a trusted party responsible for registration and revocation. Vehicles are assumed to be potentially malicious, while ENs could be compromised by attackers. The

communication channel between vehicles and ENs is vulnerable to various attacks. The paper introduces the edge-assisted decentralized authentication (EADA) architecture, which delegates the authentication capability from the RS to distributed ENs. The proposed protocol consists of two authentication scenarios: Auth-I and Auth-II. In Auth-I, vehicles are collaboratively authenticated by a subset of ENs using identity-based signature techniques. The involved ENs are efficiently authenticated in a batch by the vehicle. For Auth-II, a vehicle with a valid token can achieve fast handover authentication by utilizing the token as a private credential with the nearest EN, reducing authentication delays significantly. The evaluation of the proposed protocol performance and reports significant reductions in authentication delays. The proposed EADA architecture and threshold mutual authentication protocol address security requirements such as mutual authentication between vehicles and ENs, secure token generation, and resistance against various attacks. However, with the number of vehicles increases, this will lead to an increase in EN storage.

In the context of 5G (Fifth Generation) networks, AKA (Authentication and Key Agreement) refers to the authentication and key establishment process used to secure communication between user devices and the 5G network infrastructure. 5G - AKA [72] is an enhanced version of the authentication mechanism used in previous generations of mobile networks, such as 4G LTE (Long Term Evolution). It provides improved security and privacy features to address the evolving requirements and challenges of 5G networks. 5G AKA incorporates advanced security features, such as mutual authentication, forward secrecy, and protection against replay attacks. These enhancements ensure the confidentiality, integrity, and authenticity of the communication between the UE and the 5G network, providing a secure environment for 5G services and applications.

The 4G EPS-AKA refers to the Fourth Generation Evolved Packet System Authentication and Key Agreement [73]: It is a security protocol used in 4G LTE (Long Term Evolution) networks to authenticate and establish secure communication between mobile devices and the network. It provides mutual authentication between the user equipment (UE) and the network, ensuring that both parties can verify each other’s identities. The EPS-AKA protocol ensures the confidentiality, integrity, and mutual

authentication of data transmitted over the LTE network. It helps protect against unauthorized access, eavesdropping, and various security threats, providing a secure environment for mobile communication.

While 4G EPS-AKA and 5G – AKA provide a secure authentication and key establishment process, in an IoT (Internet of Things) environment, they have certain limitations and privacy threats[74], [75], [76], [77], [78], [79]: Power consumption: IoT devices are often battery-powered and designed for low-power operation. The authentication process in 4G EPS – AKA/5G – AKA involves multiple steps, including cryptographic operations, which can consume significant power. This can be a challenge for resource-constrained IoT devices with limited battery life. Overhead: 5G – AKA introduces additional signaling and processing overhead for authentication and key agreement. In an IoT environment with a large number of devices, this overhead can impact network efficiency and scalability. Latency: The authentication process in 4G EPS – AKA involves several round trips between the IoT device and the network, which can introduce latency. In applications where low latency is critical, such as real-time control systems, this delay may not be acceptable. User Tracking: The authentication process in 4G EPS – AKA involves the exchange of identifiers and authentication parameters between the user equipment (UE) and the network. Adversaries or service providers could potentially use this information to track users' movements and behavior patterns, compromising their privacy. Network Surveillance: During the authentication process, user devices interact with the network infrastructure. Network operators or other entities with access to network data could monitor and analyze these interactions, potentially compromising user privacy.

TLS 1.3, which stands for Transport Layer Security [55] 1.3, is a cryptographic protocol used to establish a secure and encrypted connection between a client and a server over a network. It is the latest version of the TLS protocol and offers significant improvements over its predecessor, TLS 1.2. TLS 1.3 [80] removes support for older cryptographic algorithms and cipher suites that are considered weak or vulnerable to attacks. It promotes the use of stronger encryption algorithms and ensures better security for data transmission. It also reduces the number of round trips required during the handshake

process, resulting in faster connection establishment. This is achieved by using a zero-RTT (Round Trip Time) mode, allowing the client and server to resume a previous connection without the need for a full handshake.

However, the TLS 1.3 has some limitations related to IoT environment [81], [82]. Many IoT devices, such as sensors or actuators, have limited processing power, memory, and battery life. Implementing the full TLS 1.3 protocol stack on such resource-constrained devices can be challenging. The increased complexity of TLS 1.3 compared to previous versions may require more computational resources, which could impact the overall performance of the device or drain its battery quickly. TLS 1.3 introduces a new handshake mechanism that reduces the number of round trips required to establish a secure connection. However, even with the reduced round trips, the handshake process still incurs additional overhead, which may not be suitable for low-power or latency-sensitive IoT applications. This overhead can impact the responsiveness and real-time nature of IoT devices.

2.3 Inspiration from the Previous Work

The previous works on authentication protocols in fog computing, vehicular ad hoc networks, Internet of Things (IoT), TLS 1.3, and 4G/5G networks provide valuable insights and inspiration for developing secure and efficient authentication mechanisms. These works highlight the importance of addressing the unique challenges and requirements of different computing environments, such as resource constraints, mobility, privacy concerns, and scalability. They also demonstrate the application of various cryptographic techniques, including bilinear pairings, elliptic curve cryptography (ECC), one-way hash functions, and blockchain, to achieve secure authentication while minimizing computational and communication overhead.

One key inspiration from these works is the concept of conditional privacy-preserving authentication. By disclosing only necessary information for authentication while preserving anonymity, these protocols strike a balance between security and efficiency. The TLS 1.3 protocol serves as an inspiration due to its advancements in security and performance. However, its limitations in resource-constrained IoT environments should

be carefully considered, and alternative approaches, considering the processing power, memory, and battery life of IoT devices, need to be explored.

Overall, previous works provided a foundation for designing authentication protocols that address the specific challenges and requirements of fog computing, IoT, vehicular networks, and 5G environments. They emphasize the need for secure, lightweight, and efficient authentication mechanisms that ensure privacy, scalability, and resilience against various known attacks. Building upon these inspirations, our work aims to provide a lightweight mutual authentication and session key agreement in the cloud-fog-edge three-tier architecture, which supports fast handover.

Chapter 3

3 Three-tier architecture Environment

This chapter provides an extensive exposition on both the conceptualization and modeling of the cloud-fog-edge three-tier environment. The term "three-tier environment" encompasses a distributed computing framework that encompasses cloud computing in the upper tier, fog computing in the intermediary tier, and edge computing in the foundational tier. Each of these tiers assumes a pivotal function in facilitating streamlined and scalable computation, storage, and data processing, catering to a diverse spectrum of applications. This chapter delves into the fundamental attributes and constituent elements of this architecture while also investigating scenarios that underscore the importance of authentication. Table 1 displays the notations utilized in proposed model.

Table 1: Three-tier Model Notations

Notation	Description
δ	A set of Edge devices
D	Number of Edge devices represented as $\delta = \{\delta_1, \delta_2, \ldots, \delta_d\}$ where $1 \leq d \leq D$
ζ	A set of Fog Nodes
Z	Number of Fog Nodes represented as $\zeta = \{\zeta_1, \zeta_2, \ldots, \zeta_z\}$ where $1 \leq z \leq Z$
γ	A set of Cloud Data Centers
C	Number of Cloud Data Centers represented as $\gamma = \{\gamma_1, \gamma_2, \ldots, \gamma_c\}$ where $1 \leq c \leq C$
u	A set of 5G service providers
S	A temporary secret key for each entity
K	A secret key generated by 5G service provider during the registration phase for each entity
O	A security token generated by 5G service provider during the registration phase for each entity

3.1 Environment Characteristics

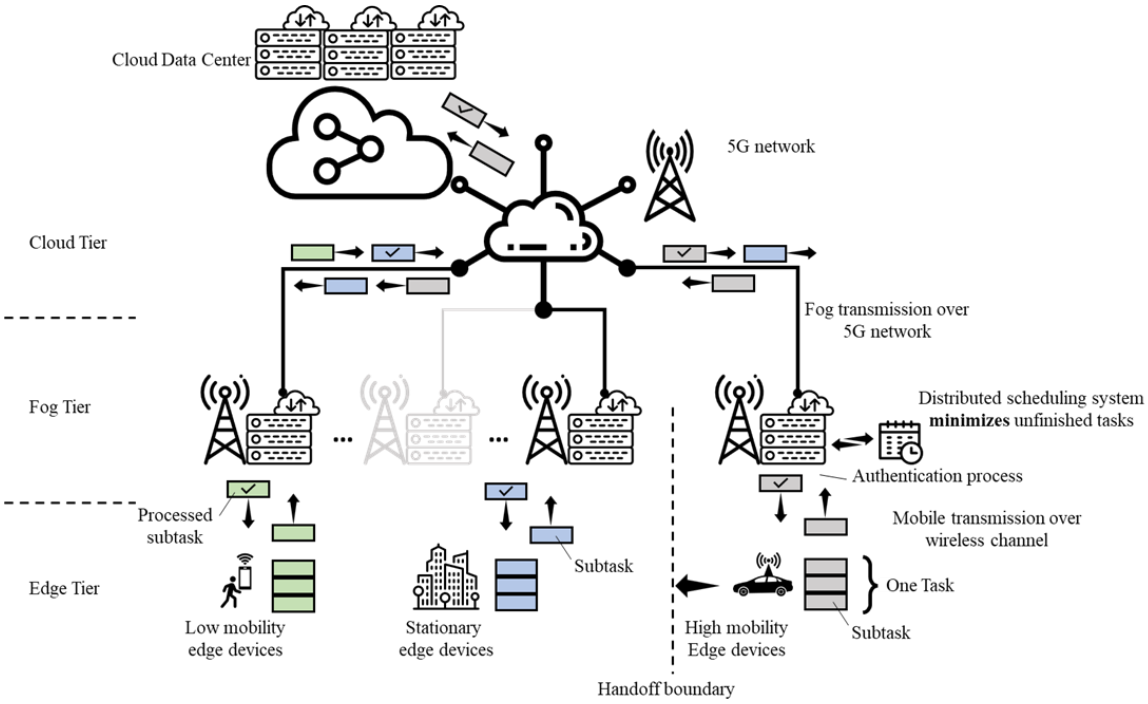

Figure 1: Three tier architecture

The diagram presented in Figure 1 above illustrates the cloud-fog-edge three-tier architecture, a sophisticated design paradigm that leverages the integration of cloud computing, fog computing, and edge computing to create a powerful and versatile system. This architecture organizes computing resources into three layers, each with distinct characteristics. The subsequent section presents an in-depth analysis of each tier, elucidating their distinct characteristics using the modeling notations shown above:

The edge tier has the following characteristics:

- Set of D mobile edge devices represented as $\delta = \{\delta_1, \delta_2, \ldots, \delta_d\}$, where $1 \leq d \leq D$.
- Mobility: edge devices can move from one place to another [83].
- Interoperability: Edge devices may depend on its operation with other heterogeneous devices and service architectures. The edge tier exhibits

heterogeneity due to variations in device architectures, communication protocols, and network configurations [83].

- Scalability: The scalability of the edge tier depends on the number of mobile users, various applications, and low-bandwidth networks. It can be scaled by expanding geographically, adding new service nodes to existing locations, and utilizing cloud interaction [83].

The Fog tier has the following characteristics:

- Set of Z Fog nodes represented as $\zeta = \{\zeta_1, \zeta_2, \ldots, \zeta_z\}$, where $1 \leq z \leq Z$.
- Low latency and real time interactions: Fog nodes close to the network edge collect, process, and store sensor and device data. This enables low latency and meets the needs of real-time interactions, particularly for latency-sensitive or time-sensitive applications [22].
- Save bandwidth: Certain computation tasks, such as data preprocessing, redundancy removal, data cleaning, and filtering, are executed locally. Data processing is distributed across fog nodes rather than relying solely on centralized cloud servers. This distribution improves scalability and reduces the load on the central cloud infrastructure. Only relevant data is sent to the cloud, minimizing unnecessary data transmission over the Internet [22].
- Heterogeneity [22]: Fog nodes are available in various form factors and can be deployed as physical or virtual nodes in diverse environments. They encompass high-performance servers, edge routers, gateways, access points, base stations, and more. These hardware platforms exhibit distinct levels of computation and storage capabilities, run various operating systems (OS), and support different software applications [22].
- Interoperability: Fog nodes and end devices, sourced from different providers and deployed in diverse environments, possess heterogeneous characteristics. To effectively handle a broad array of services and ensure smooth support, fog computing necessitates interoperability and cooperation with various providers. This includes interoperation among multiple fog nodes and devices within the same fog, as well as interoperability with cloud computing [22].

The cloud tier has the following characteristics:

- Set of C cloud data centers represented as $\gamma = \{\gamma_1, \gamma_2, \ldots, \gamma_c\}$, where $1 \leq c \leq C$.
- Service oriented: The service-oriented concept is a practical alternative to Service Oriented Architecture (SOA). Cloud computing services are categorized into Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS), and Software-as-a-Service (SaaS). Users can utilize these services without owning hardware or knowing data locations. Examples include cloud storage, Google App Engine, and online gaming [21].
- Scalability: Server, resources, client. Clouds offer the automatic resizing of virtualized hardware resources. Scalability requires dynamic reconfiguration: as the system scales it needs to be reconfigured in an automated manner [84].
- On-Demand Self-Service: Users can provision computing resources (such as virtual machines, storage, and applications) as needed, without requiring human intervention from the service provider [85].
- Broad Network Access: Cloud resources are accessible over the network and can be accessed by various devices with internet connectivity, such as laptops, smartphones, and tablets [85].
- Resource Pooling: Cloud providers use multi-tenant models to pool and share resources among multiple users, with resources dynamically allocated based on demand. Users typically do not have control over the exact physical location of the resources [85].
- Rapid Elasticity: Cloud resources can be scaled up or down quickly to accommodate changes in demand. This elasticity allows users to access additional resources during peak periods and release them when they are no longer needed [85].
- Measured Service: Cloud systems automatically monitor and track resource usage, enabling users to be billed based on their consumption. This pay-as-you-go model promotes cost efficiency and resource optimization.

The entities within the 5G network environment have the following assumptions:

- The 5G network backhaul establishes a connection between fog nodes and the cloud data center. To ensure secure communication, a temporary secret key is assigned for the cloud initial entry into the environment.
- Each edge device and fog node have eSIM and a temporary secret key *S* stored in eSIM.
- After the registration phase, the entity (i.e., Edge device, Fog node, or Cloud server) will get a secret key *K* and a token *O* which are generated by 5G service provider u_v.

3.2 Authentication within the environment

Within the cloud-fog-edge three-tier environment, various communication scenarios occur, involving direct communication between edge devices and fog nodes and interactions between fog nodes themselves and between fog nodes and the cloud. These communication channels must be secured to protect sensitive data and maintain the system's integrity [9].

Authentication plays a crucial role in the cloud-fog-edge computing environment due to several key needs and challenges:

- Data Security: In a distributed architecture like cloud-fog-edge, data is processed and stored across various tiers, from edge devices to fog nodes and cloud servers. Authentication ensures that only authorized entities can access and manipulate sensitive data, reducing the risk of unauthorized data breaches or leaks [86].
- Resource Access Control: Different tiers of the architecture have varying levels of resources and capabilities. Authentication helps in controlling access to these resources based on user roles, permissions, and the specific needs of applications, ensuring optimal resource utilization [86].
- Seamless Handovers: Devices moving between different tiers, such as from edge to fog or from fog to cloud, require smooth handovers without interrupting services. Authentication enables seamless handovers by ensuring that the device is authenticated in the new environment before resuming communication [59].

- Dynamic Mobility: Devices in this environment can be highly mobile, moving across different tiers and networks. Authentication mechanisms are needed to handle dynamic mobility patterns without disrupting services [87].
- Latency and Real-Time Requirements: Some applications in fog-edge environments have stringent latency requirements. Authentication mechanisms must be fast and efficient to avoid introducing unacceptable delays [88].
- Adversarial Threats: Distributed systems are susceptible to various threats, including man-in-the-middle attacks and impersonation. Authentication mechanisms need to counteract these threats effectively [86].
- Authentication Overhead: Introducing authentication processes can lead to communication overhead. Balancing the need for security with the performance impact of authentication is a challenge [71].

To accomplish the need and overcome challenges, a mutual entity authentication protocol is required.

The proposed mutual authentication protocol serves as a crucial security mechanism within the environment. It ensures that the communicating entities, such as edge devices, fog nodes, and the cloud, authenticate each other's identities before initiating any data exchange. This mutual authentication process establishes trust and prevents unauthorized access or malicious attacks, safeguarding the integrity and confidentiality of the communication channels.

Given the mobile nature of certain edge devices, a mutual security protocol specifically designed for the handover process is necessary. When an edge device transitions to a new fog node, it needs to undergo mutual identity verification to establish a secure connection. This verification step ensures that the edge device and the new fog node mutually authenticate each other's identities before enabling communication. By verifying the identities of both parties involved, the protocol guarantees that only authorized edge devices can connect to new fog nodes, reducing the risk of unauthorized access and potential security vulnerabilities.

The proposed implementation of a mutual identity verification protocol for handover in the cloud-fog-edge three-tier environment reinforces the security measures and addresses the unique challenges posed by mobile edge devices. It enables seamless and secure communication, ensuring uninterrupted access to services and resources during the handover process. This enhances the environment's overall reliability, efficiency, and integrity, simplifying the process of moving between fog nodes without causing noticeable interruptions or disruptions.

Chapter 4

4 Proposed Approach

This chapter comprehensively presents the proposed edge-fog-cloud three-tier mutual authentication protocol. The authentication protocol is structured into four main phases:

1. Initialization: mainly responsible for generating the temporary secret key and master key.
2. Registration: identifies edge devices, fog nodes, and cloud data centers within the three-tier architecture
3. Authentication and key agreement: ensures the verification process between edge devices, fog nodes, and cloud data centers, and facilitates the creation and distribution of session keys for secure communication.
4. Handover verification: focuses on validating the edge user and the new fog node during the handover process. This phase ensures seamless and secure transitions during handovers.

Table 2 compliments the modeling notations we presented in table 1 and displays the extended notations utilized in the protocol design.

Table 2: Protocol Notation

Notation	Description
S_{δ_d}	A temporary secret key S for edge device δ_d
S_{ζ_z}	A temporary secret key S for fog node ζ_z
S_{γ_c}	A temporary secret key S for cloud data center γ_c
K_{δ_d}	A secret key K generated by 5G service provider during the registration phase for edge device δ_d
K_{ζ_z}	A secret key K generated by 5G service provider during the registration phase for fog node ζ_z
K_{γ_c}	A secret key K generated by 5G service provider during the registration phase for cloud data center γ_c
O_{δ_d}	A security token O generated by 5G service provider during the registration phase of the edge device δ_d
O_{ζ_z}	A security token O generated by 5G service provider during the registration phase of the fog node ζ_z

O_{γ_c}	A security token O generated by 5G service provider during the registration phase of the cloud data center γ_c
MS_{u_v}	The master secret key of the 5G service provider u_v
T	A timestamp T is the current time that both participants record during the session.
$SK_{\delta_d \zeta_z}$	A generated session key to be used by the edge device δ_d and the fog node ζ_z
$SK_{\delta_d \gamma_c}$	A generated session key to be used by the edge device δ_d and the cloud data center γ_c
$SK_{\zeta_z \gamma_c}$	A generated session key to be used by the fog node ζ_z and the cloud data center γ_c
R_{δ_d}	Response message from 5G service provider to edge device δ_d
R_{ζ_z}	Response message from 5G service provider to fog node ζ_z
R_{γ_c}	Response message from 5G service provider to cloud data center γ_c
$M_{\delta_d \zeta_z}$	Messages between edge device δ_d and fog node ζ_z
$M_{\delta_d \gamma_c}$	Messages between edge device δ_d and cloud data center γ_c
$M_{\gamma_c \zeta_z}$	Messages between fog node ζ_z and cloud data center γ_c
E (p, k)	The encryption of the plaintext p with the encryption key k using the AES-128 encryption technique.
D (c, k)	The decryption of the ciphertext c with the encryption key k, using the AES-128 encryption technique.
$\|\|$	Concatenation: Combining or linking two or more strings, sequences, or values together in a specific order to create a longer sequence or string

4.1 Proposed Authentication protocol

This section introduces the four phases within the authentication protocol. It includes the initialization, registration, authentication, key agreement, and handover verification phases.

4.1.1 Initialization phase

In the initialization phase, each entity (the edge devices δ_d , fog nodes ζ_z , and cloud data centers) request eSIM from 5G service provider u_v. The eSIM of the edge device includes the device ID (δ_d ID) and a temporary secret key S_{δ_d}. The fog node's eSIM includes the node ID (ζ_z ID) and a temporary secret key S_{ζ_z}. Similarly, the cloud data

center's eSIM includes its ID ($\gamma_c\ ID$) and a temporary secret key S_{γ_c}. The 5G service provider u_v maintains a master secret key MS_{u_v}, this key will never leave the u_v and there will be one master key per application. Note that the 5G service provider also knows these temporary secret keys.

4.1.2 Registration phase

In the registration phase, an entity registers to the 5G service provider. This entity can be an edge device, fog node or cloud data center. Figure 2 shows the registration protocol of the edge device δ_d. The following steps describe the details of this protocol. Note that, the registration protocols for the fog nodes and the cloud data centers should follow the same protocol steps.

Step 1. The edge device starts the session by sending a hello message, including the device ID ($\delta_d\ ID$) to register into the 5G service provider u_v, as shown in line 1 in Figure 2.

Step2. The 5G service provider u_v receives the hello message, generates a secret key K_{δ_d} and computes a security token O_{δ_d}. This token is the encryption (using AES) of the device ID along with the generated secret key K_{δ_d} using the master key of the service provider MS_{u_v}, denoted by $E("\delta_d ID" \,||\, K_{\delta_d}, MS_{v_v})$ as shown in line 2 in Figure 2. In line 3, u_v computes a response message R_{δ_d} = E ($K_{\delta_d} \,||\, O_{\delta_d}, S_{\delta_d}$), to send the generated secret key and the token to the edge device, where S_{δ_d} is the temporary secret key of the edge device.

Step3. The edge device δ_d receives the response message R_{δ_d} from 5G service provider u_v . δ_d decrypts the R_{δ_d} to retrieve and store the secret key K_{δ_d} and the token O_{δ_d} , using its temporary secret key S_{δ_d} as follows; $D(K_{\delta_d} || O_{\delta_d}, S_{\delta_d})$. Then, in line 4 in Figure 2, the edge device forgets the temporary secret key S_{δ_d} to prevent future replay attacks, and the 5G service provider forgets the temporary and generated device secret keys to keep the protocol stateless as much as possible and to prevent denial-of-service attacks.

This step ends the registration phase, where the edge device possesses a permanent secret key and a sealed security token that can be opened only by the 5G service provider. In addition, the service provider does not store any information about the registered edge device.

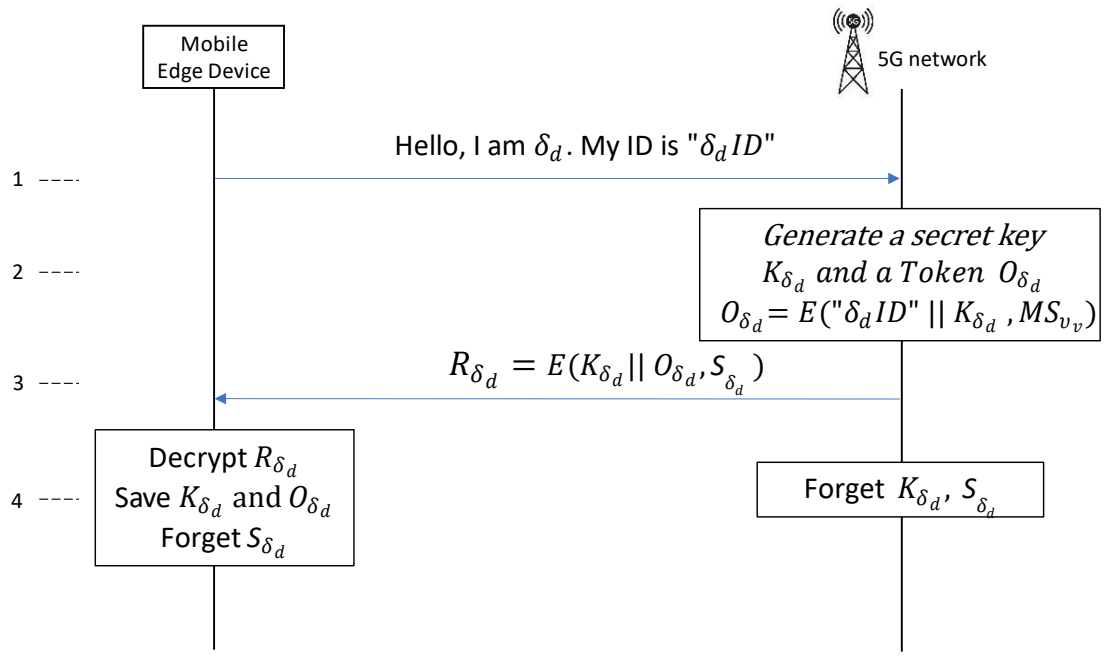

Figure 2: Entity Registration Protocol

4.1.3 Authentication and Key Agreement phase (Edge device and Fog node Authentication Protocol)

In this phase, if a registered edge device δ_d is going to join the network, the authentication and key agreement process between edge device δ_d, fog node ζ_z and 5G service provider u_v will be performed. Figure 3 shows the steps of the authentication and key agreement protocol to accomplish mutual authentication between fog nodes and edge devices and generate a common session key for future security services.

Step 1. The edge device δ_d computes a self-authentication message *proofMe* , which is the encryption of the current timestamp *T* along with a string literal "$\delta_d to \zeta_z$" using the

edge secret key K_{δ_d}, $proofMe = E(T \,||\, "\delta_d to \zeta_z",\, K_{\delta_d})$. The timestamp ensures the refresh of the message and prevent the replay attacks, whereas the string literal to request from the 5G service provider that the edge device δ_d would like to communicate with the fog node ζ_z . Then δ_d transmits *proofMe* and the security token O_{δ_d} to 5G service provider u_v. See lines 1 and 2.

Step 2. 5G service provider u_v receives the self-authentication message *proofMe* and the token O_{δ_d} from edge device δ_d. Then u_v decrypts the received token to recognize the identity of the edge device, i.e., the δ_dID, and to retrieve the secret key of δ_d , i.e., K_{δ_d}. u_v decrypts the received *proofMe* using K_{δ_d}, and verifies that *T* is within the current time skew. If the timestamp is verified, the δ_d is authenticated to u_v who will generate a session key $SK_{\delta_d \zeta_z}$ to be used between edge device δ_d and fog node ζ_z. However, if the timestamp verification failed, u_v closes the session. Please see lines 2 and 3. In lines 4 and 5, u_v encrypts the string "$\delta_d to \zeta_z$" along with the generated session key using their master secret key MS_{u_v}, which we called $proof_{\delta_d}$. Note that this $proof_{\delta_d}$ is a sealed value of the generated session key $SK_{\delta_d \zeta_z}$ that will not be saved in the 5G server provider, and again, to make it a stateless entity. As shown in line 5, the retrieved timestamp, the string "$\delta_d to \zeta_z$" , the session key $SK_{\delta_d \zeta_z}$, and the $proof_{\delta_d}$ are encrypted by the secret key K_{δ_d} to form the message response R_{δ_d} , i.e., $R_{\delta_d} = E(T||"\delta_d to \zeta_z"||SK_{\delta_d \zeta_z}||proof_{\delta_d},\, K_{\delta_d})$. Then u_v sends R_{δ_d} to the edge device δ_d .

Step 3. Once edge device δ_d receives the response message R_{δ_d} , as shown in line 5, it decrypts R_{δ_d} , and first verify the value of the timestamp to authenticate the 5G service provider u_v who verified the device δ_d and created a secret session key to be used between δ_d and the fog node ζ_z. Note that, the value $proof_{\delta_d}$ was sent to the edge device not to the fog node, to avoid involving the fog node in the protocol until the edge device decide to do so. Now, the edge device decrypts R_{δ_d} to retrieve the session key $SK_{\delta_d \zeta_z}$ and $proof_{\delta_d}$. Then it stores $SK_{\delta_d \zeta_z}$ in its memory and computes a message to fog node ζ_z: $M_{\delta_d \zeta_z} = E\,(T \,||\, "\delta_d to \zeta_z",\, SK_{\delta_d \zeta_z})$, where *T* is the timestamp at the current time of line 6. Finally, δ_d transmits $M_{\delta_d \zeta_z}$ and $proof_{\delta_d}$ to ζ_z.

Step 4. Fog node ζ_z receives the message $M_{\delta_d \zeta_z}$ and $proof_{\delta_d}$ from edge device δ_d to indicate that an edge device wishes to connect with the fog node ζ_z. At this point, the fog node will depend on the 5G service provider u_v to validate this request and get a secret key to be used between this edge device and the fog node. Therefore, the fog node will authenticate itself to the u_v by sending a self-authentication message $proofMe = E\ (T,\ K_{\zeta_z}\ )$ along with the value $proof_{\delta_d}$ and the token O_{ζ_z} as shown in lines 7 and 8.

Step 5. The 5G service provider receives the token O_{ζ_z} by which it recognize the sender who is ζ_z , it uses the secret key K_{ζ_z} to verify the received $proofMe$ and authenticate the fog node. Then u_v decrypts the received $proof_{\delta_d}$, check the text "$\delta_d to \zeta_z$" , and retrieve the session key $SK_{\delta_d \zeta_z}$. Finally, u_v computes a response message $R_{\zeta_z} = E\ (T\ ||\ SK_{\delta_d \zeta_z}, K_{\zeta_z}\ )$ and transmits R_{ζ_z} to ζ_z. as shown in lines 9 and 10. Note that, u_v will forget O_{δ_d}, O_{ζ_z}, K_{δ_d}, K_{ζ_z}, $SK_{\delta_d \zeta_z}$, $proofMe$ and $proof_{\delta_d}$ after sending R_{ζ_z}.

Step 6. Once fog node ζ_z received the response message R_{ζ_z} from 5G service provider u_v, it decrypts R_{ζ_z} , verify the value of the timestamp to authenticate the 5G service provider u_v, retrieve the session key $SK_{\delta_d \zeta_z}$, and stores it in its memory. Then, in line 11, ζ_z decrypts the received message $M_{\delta_d \zeta_z}$ (in line 6) to recognize the string "$\delta_d to \zeta_z$" and the identity of the requesting edge device, who was δ_d . At this point ζ_z authenticates δ_d . Now, in order for ζ_z to authenticate itself to δ_d , computes and sends the message $M_{\zeta_z \delta_d} = E\ (T\ ||\ "\zeta_z to \delta_d",\ SK_{\delta_d \zeta_z})$ to δ_d.

Step 7. The edge device δ_d decrypts the received message $M_{\zeta_z \delta_d}$, using the session key $SK_{\delta_d \zeta_z}$, as shown in line 12. The edge device δ_d authenticates ζ_z if the string literal is "$\zeta_z to \delta_d$" and the retrieved timestamp T is within the time skew of the timestamp at line 6. If successful, the mutual authentication and key agreement process is completed. Note that the real identity of the edge device and the fog node were concealed during this protocol.

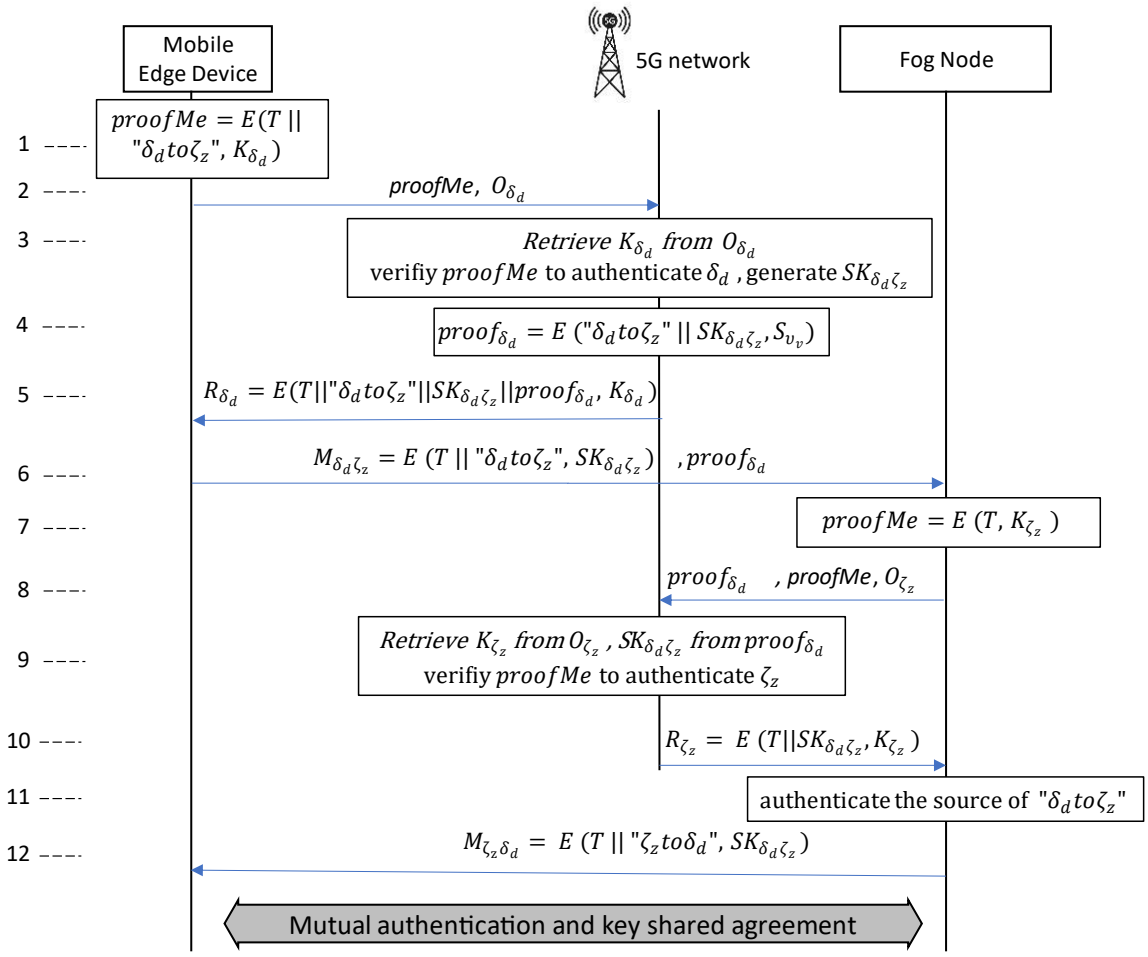

Figure 3: Authentication and Key Agreement Protocol (Edge device and Fog node)

4.1.4 Authentication and Key Agreement phase (Fog node and Fog node Authentication Protocol)

During this phase, a fog node ζ_1 aims to establish a secure communication channel with another fog node ζ_2 with the presence of the 5G service provider u_v.

The structure of the fog node to fog node authentication and key agreement protocol closely resembles the structure of the edge device to fog node protocol described above. Therefore, we will exclusively present a graphical representation for the fog node to fog node protocol (in Figure 4) depicting its methodology, accompanied by comprehensive annotations for each procedural step, obviating the need for textual elucidation.

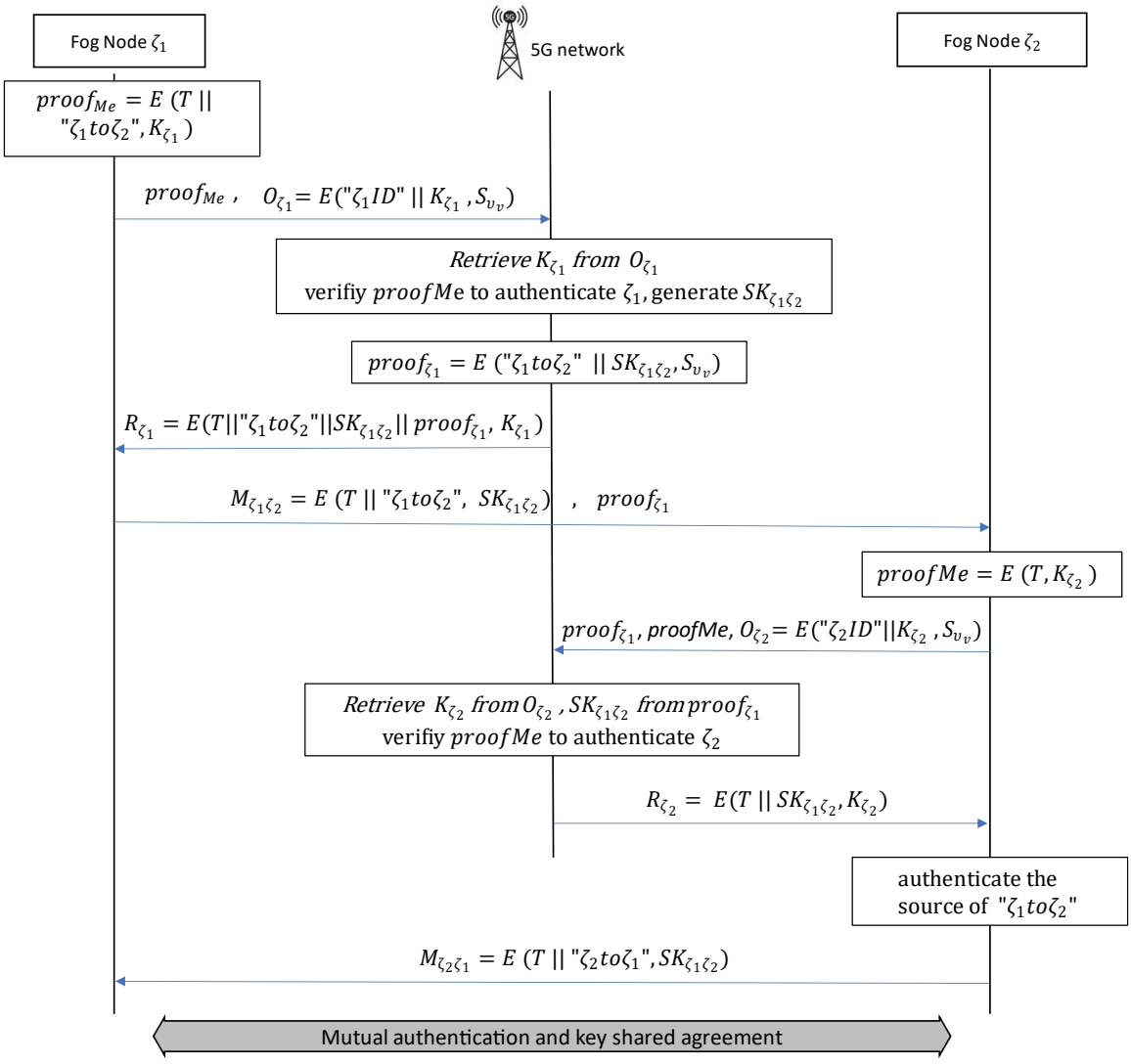

Figure 4: Authentication and Key Agreement phase (Fog node and Fog node)

4.1.5 Authentication and Key Agreement phase (Fog node and Cloud Authentication Protocol)

In this phase, a fog node ζ_z endeavors to establish a secure communication channel with a cloud data center γ_c with the presence of the 5G service provider u_v.

The structure of the fog node to cloud data center authentication and key agreement protocol closely resembles the structure of the edge device to fog node protocol described above. Therefore, we will exclusively present a graphical representation for the fog node to cloud data center protocol (in Figure 5) depicting its methodology, accompanied by comprehensive annotations for each procedural step, obviating the need for textual elucidation.

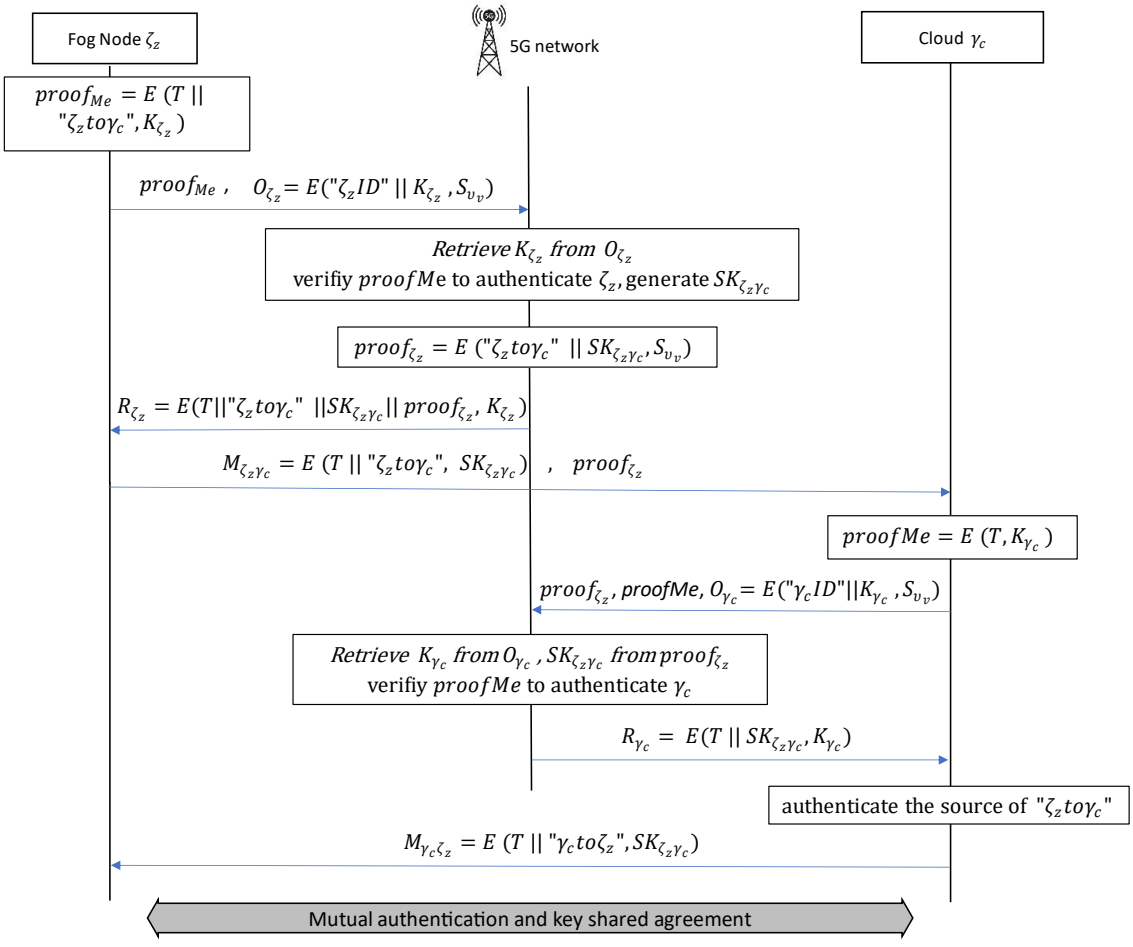

Figure 5: Authentication and Key Agreement phase (Fog node and Cloud)

4.1.6 Edge Device Handover Authentication phase

During this phase, when an edge device δ_d relocates from one fog node ζ_1 to another, a fog node ζ_2, a handover process is initiated. The edge device δ_d undergoes mutual authentication with the new fog node ζ_2. To establish mutual authentication between the target fog nodes and edge devices, alongside the creation of a shared session key for upcoming security services, as illustrated in Figure 6, the subsequent steps delineate the specifics of this authentication and key agreement protocol.

Step 1. The current fog node ζ_1 performs the fog node to fog node mutual authentication protocol and generates a secure session key $SK_{\zeta_1\zeta_2}$. See line 1 and Figure 4.

Step 2. Fog node ζ_1 generates a secure random key denoted as $SK_{\delta d\zeta_2}$, intended for utilization by the target fog node ζ_2 and edge device δ_d. It then proceeds to generate two handover messages, namely $M_{\zeta_1\zeta_2}$ and $M_{\zeta_1\delta d}$. The content of message $M_{\zeta_1\zeta_2}$ consists of the encryption of the present timestamp *T* along with a string literal "$\zeta_1 to \zeta_2$" and $SK_{\delta d\zeta_2}$ using the secret key $K_{\zeta_1\zeta_2}$, $M_{\zeta_1\zeta_2} = E(T \| \text{"}\zeta_1 to \zeta_2\text{"} \| SK_{\delta d\zeta_2}, SK_{\zeta_1\zeta_2})$. Analogously, the message $M_{\zeta_1\delta d}$ encompasses the encryption of the current timestamp *T* along with a string literal "$\zeta_1 to \delta d$" and $SK_{\delta d\zeta_2}$ utilizing the secret key $SK_{\zeta_1\delta d}$, $M_{\zeta_1\delta d} = E(T \| \text{"}\zeta_1 to \delta d\text{"} \| SK_{\delta d\zeta_2}, SK_{\zeta_1\delta d})$. Subsequently, ζ_1 transmits $M_{\zeta_1\zeta_2}$ to ζ_2 and $M_{\zeta_1\delta d}$ to δ_d. Then ζ_1 forgets $SK_{\delta d\zeta_2}$. See lines 2, 3 ,4 and 5.

Step 3. Edge device δ_d and the target fog node ζ_2 receives the handover message from the fog node ζ_1. Then both δ_d and ζ_2 possess a shared session key $SK_{\delta d\zeta_2}$ to establish a secure channel.

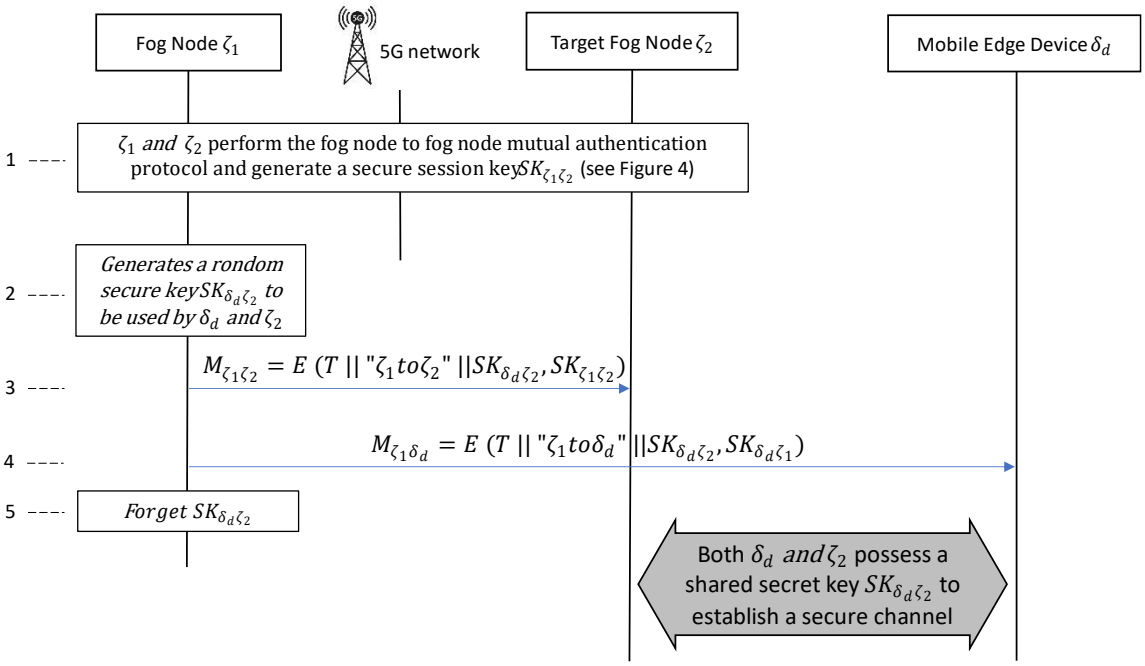

Figure 6: Handover Authentication phase

Chapter 5

5 Security and feature analysis

The security and feature analysis refers to evaluating and examining the security aspects and functional features of a particular system, technology, or software. It involves assessing the system's strengths, weaknesses, vulnerabilities, and capabilities from both a security and functional perspective. This chapter shows the security and feature analysis of the three-tier mutual authentication protocol.

5.1 Security analysis

This analysis focuses on identifying and evaluating potential security risks, threats, and vulnerabilities associated with the system. It involves assessing the effectiveness of security measures implemented within the system when facing attacks, such as spoofing attacks, information disclosure, denial of service, and elevation of privilege. The security analysis aims to identify any weaknesses or potential areas of exploitation that could compromise the system's security.

5.1.1 Data integrity and Tampering attack

Data integrity in an authentication protocol refers to the assurance that the transmitted or stored data remains unchanged and uncorrupted throughout the authentication process. It ensures that the data has not been tampered with or modified in any unauthorized manner. Data integrity is crucial in authentication protocols because any alteration or manipulation of data can lead to security breaches or unauthorized access.

A tampering attack refers to the unauthorized modification of the messages moving back and forth between two participants in the target protocol, with the aim of changing the original message to achieve some malicious purpose. This protocol is designed to prevent message leakage. Each message sent between participants is encrypted. Once the attacker modifies the plaintext sent among the protocol participants, the receiver cannot decrypt it with the specified secret key. This modification will be discovered immediately.

The following scenarios depict the potential consequences when a message undergoes modifications during authentication and key agreement phase message transmission.

In step1, if token O_{δ_d} is modified, the 5G service provider u_v will not be able to retrieve the accurate secret key K_{δ_d} due to it being encrypted by u_v master secret key S_{u_v}. If proofMe is modified, u_v cannot get the time stamp and compare it to the current time stamp, or get the correct user ID. Moreover, it is highly improbable for an adversary to create a legitimate O_{δ_d} since its creation involves the utilization of S_{u_v}, which represents the master key of the 5G service provider.

During step 2, it is important to note that the adversary cannot have knowledge of the edge user's secret key K_{δ_d}. Therefore, any attempts to modify "$\delta_d to \zeta_z$", $K_{\delta_d \zeta_z}$, or proofδ_d would be infeasible for the adversary. Similar to step2, the adversary is unable to tamper "$\delta_d to \zeta_z$", or timestamp in step 3.

Suppose the adversary modifies the transmission message in the fourth step. In that case, the 5G service provider will be unable to retrieve any valid information due to the use of master secret key. Then, the authentication will fail.

It is crucial to emphasize that safeguarding data integrity is not solely reliant on the authentication protocol but is a vital aspect to be considered across the entire system's design and implementation. In our designed protocol, we have incorporated AES encryption technology to encrypt every transmitted message between entities, thereby demonstrating the protocol's ability to ensure data integrity and prevent tampering attacks.

5.1.2 Spoofing

Spoofing is a type of security attack where an attacker pretends to be a specific edge device to deceive the fog node (the victim) into revealing the keying information, or vice versa. In this protocol, if a malicious attacker *M* wants to pretend to be a legitimate user, *M* must calculate a self-authentication message $proofMe = E(T \,||\, "\delta_m to \zeta_z",\ K_{\delta_m})$ and a token $O_{\delta_m} = E("\delta_m ID" \,||\, K_{\delta_m}, S_{v_v})$, where K_{δ_m} is a faked secret key for M and S_{u_v} is the master key of the 5G service provider. The attacker can generate a faked proofMe

message, but not the token, because S_{u_v} is known only by the 5G service provider. Therefore, the attacker will send a token that is recorded from previous sessions as a fake token instead. However, the 5G network will drop and close the session because the retrieved secret key in the received token does not match the attacker faked secret key for M. Therefore, the proposed protocol can prevent spoofing attacks.

5.1.3 Man-In-the-Middle attacks

A Man-in-the-Middle (MitM) attack is a cybersecurity attack where an attacker intercepts and potentially alters communications between two parties without their knowledge. This type of attack occurs when an attacker positions themselves between the sender and receiver of data, allowing them to capture, manipulate, or eavesdrop on the information being exchanged. In this protocol, should a malicious actor labeled as *M* captures the message sent from an edge device to a 5G network and want to get the data, the acquisition of the master key S_{v_v} becomes imperative. This key serves as the mean to decrypt authentication messages and any additional messages intended for *M*. The absence of said master key renders M incapable of accessing any information. The response message sent from the 5G service provider to edge device δ_d is encrypted by the secret key of δ_d. If *M* does not know the secret key, *M* is unable to access or alter the data. The session key encrypts messages transmitted between edge device and the fog node $K_{\delta_d \zeta_z}$. For *M* to illicitly acquire the data after intercepting the information, it is imperative that *M* possesses the session key. The above scenarios illustrate that the designed protocol can resist MitM attack.

5.1.4 Replay attacks

A replay attack is a type of cybersecurity attack where an attacker intercepts and then maliciously re-transmits data that was previously captured, without altering the data itself. The goal of a replay attack is not to manipulate the content of the data, but rather to cause undesirable effects by repeating legitimate data transmissions. This can lead to various security vulnerabilities and compromise the integrity and authenticity of a system. The proposed protocol is designed to guarantee the freshness of messages. For example, in the mutual authentication protocol between the edge device and fog node, 5G service

provider u_v receives the self-authentication message *proofMe* and the token O_{δ_d} from edge device δ_d. Then u_v decrypts the received *proofMe* using K_{δ_d}, and verifies that *T* is within the current time skew. The message $M_{\delta_d\zeta_z}$ includes a text "$\delta_d to \zeta_z$". This text ensures that this message's direction is from the edge device δ_d to fog node ζ_z. The malevolent attacker is unable to dispatch the message $M_{\delta_d\zeta_z}$ to other fog node in order to execute a replay attack. Therefore, the proposed protocol can prevent replay attacks.

5.1.5 Information disclosure

An information disclosure attack (also known as information leakage or information exposure) is a type of security attack where an attacker gains unauthorized access to the protocol information (messages), which can include personal data, or intellectual property. The attacker may then use this information to launch further attacks, such as phishing emails or identity theft. The proposed protocol is designed to hide the identity of the participants, which includes the edge device, the 5G network, fog node and the cloud server. Therefore, the attacker is unaware about who is exchanging the protocol messages. Also, all messages sent between entities is encrypted by Advanced Encryption Standard (AES) encryption technology. The adversary cannot guess edge/fog/cloud secret key (K_{δ_d}, K_{ζ_z} , K_{γ_c}) or 5G network master secret key S_{u_v} to illegally obtain user ID or session key.

5.1.6 Denial of service

A Denial of Service (DoS) attack is a type of security attack where the attacker attempts to disrupt the normal functioning of a targeted system, i.e., the 5G network, the fog node, or the cloud server, by overwhelming it with traffic, requests, or other types of data. The most common situation that could make a DoS attack more likely to occur if the targeted system requires saving some data in its storage to achieve its communication protocol. Regardless of the available storage size, high, or extremely high, request will eventually consume the available storage and it may become vulnerable to a DoS attack to make the targeted system unusable or unavailable. The proposed protocol is designed to eliminate the need of remembering the edge devices' IDs or their keys and remain stateless, which eliminates the possibility of the above attack.

5.1.7 Elevation of privilege

An elevation of privilege (EOP) attack is a type of security exploit where an attacker gains access to a system, i.e., the proposed protocol, that they are not authorized to access. This attack aims to escalate their privileges from a low-level user, i.e., edge device's account, to a higher level, i.e., 5G administrator role, allowing them to gain access to sensitive data such as the 5G network master secret key. This means a legitimate edge device could perform this attack by exploiting vulnerabilities in the 5G security system to gain administrative access to a system and reveal information about the 5G service provider υ_v master key. Therefore, the security level of the proposed protocol to protect against the EOP solely depends on the security level of the 5G network.

5.2 Feature analysis

This analysis examines the functional features and capabilities of the system. It involves assessing the system's intended functionality, usability, performance, scalability, and compatibility. The feature analysis aims to evaluate how well the system meets the desired requirements and objectives, and it may involve comparing the system's features with similar existing solutions or industry standards.

5.2.1 Hidden identities anonymous

Hidden identities anonymous authentication protocol is a security mechanism designed to enable authentication while preserving the privacy and anonymity of the entities involved. It allows users to authenticate themselves without revealing their actual identities to the public. This protocol is particularly useful in scenarios where privacy and anonymity are crucial, such as online transactions, communication platforms, and anonymous voting systems. The main objective of the hidden identities anonymous authentication protocol is to ensure that the authentication process does not disclose sensitive information about the users' identities. In 3Tier - AKA, during the authentication phase, the message sent from the edge device δ_d to 5G service provider u_v includes $proofMe = E(T \,||\, "\delta_d to \zeta_z",\, K_{\delta_d})$ and a token $O_{\delta_d} = E("\delta_d ID" \,||\, K_{\delta_d}, S_{\upsilon_v})$. In this message, the identity of δ_d and target fog node ζ_z are encrypted by AES encryption technology. The malicious attacker *M* cannot acquire any δ_d or ζ_z identity information from the

intercepted message. Based on this, the protocol furnishes functions for concealing user identities and enabling anonymity.

5.2.2 Mutual Authentication

Mutual authentication protocol is a security mechanism designed to establish trust and authenticate the identities of both communicating entities in a bidirectional manner. Unlike one-way authentication, where only one entity authenticates the other, mutual authentication ensures that both entities verify each other's identities, thereby establishing a secure and trustworthy communication channel. The main goal of a mutual authentication protocol is to prevent unauthorized entities from masquerading as legitimate counterparts and to mitigate the risk of man-in-the-middle attacks. By requiring both entities to prove their identities, the protocol establishes a higher level of assurance and protects against impersonation and unauthorized access. 3Tier – AKA has been designed to provide mutual authentication between the edge node and the 5G service provider, between the fog node and the 5G service provider, and between the edge device and the fog node. In Figure 3 - line 3, the 5G service provider extracts the edge device key from the received token and decrypts the received *proofME* to check the value of *T*. If *T* matches, the current time clock is authenticated. In line 5, the edge device receives and decrypts R_{δ_d}, if the retrieved *T* matches the sending *T* in line 1, then the edge device authenticate the 5G service provider. Similarly, in line 9, the 5G server provider authenticates the fog node, and in line 10, when the fog node decrypts R_{ζ_z} and finds the retrieved *T* matches the sending *T* in line 7, then the fog node authenticates the 5G service provider. In line 11, the fog node ζ_z retrieves the session key $K_{\delta_d\zeta_z}$ = D (R_{ζ_z}, K_{ζ_z}) to be used to decrypt and verify the message $M_{\delta_d\zeta_z}$ received in line 6. Again, the correct value of *T* and the text "$\delta_d to \zeta_z$" is enough to authenticate the edge device. The edge device also authenticates the fog node when it decrypts the message $M_{\zeta_z\delta_d}$ in line 12 and verify the value of *T* and the text "$\zeta_z to \delta_d$".

5.2.3 Lightweight

A lightweight authentication protocol is a security mechanism developed to verify the identity of entities engaged in communication or interaction, while minimizing the

computational and communication overhead. Its design specifically tackles the limitations imposed by resource-constrained devices and networks, including constraints such as limited processing power, memory, energy resources, and restricted bandwidth or communication range. The designed protocol considers the inherent diversity of IoT devices in the environment, recognizing that each device is unique. The following points outline the reasons why the protocol is considered lightweight. Storage resources: In 3Tier - AKA, after the registration phase and authentication phase, the 5G service provider u_v forgets every temporary key or secret key except its master key. Edge device δ_d and fog node ζ_z also forgets its temporary secret key and only stores the session key $K_{\delta_d \zeta_z}$ and secret key. Computational resources: During the registration phase, δ_d only computes once when it retrieves the secret key K_{δ_d}. Service u_v only computes once for generating the secret key and the token. During the authentication and key agreement phase, δ_d only needs to operate two AES encoding/decoding algorithms. u_v takes three and ζ_z takes two encoding/decoding algorithms. The lightweight is achieved in this protocol.

5.2.4 Generate session key

In the field of computer security and cryptography, a session key refers to a temporary cryptographic key that is generated and used during a single communication session between two entities, such as a client and a server. The session key is designed to provide secure and confidential communication by encrypting and decrypting the data exchanged between the entities. The session key is typically established through a process called key exchange or key establishment protocol. This protocol involves a series of cryptographic algorithms and techniques to securely generate and exchange the session key between the entities. After the authentication and key agreement phase, edge device δ_d, and fog node ζ_z will have the same session key $K_{\delta_d \zeta_z}$. In the authentication and key agreement (Please see Figure 3) line 5, δ_d receives a response message R_{δ_d} from service provider u_v. Then δ_d retrieves $K_{\delta_d \zeta_z}$ by performing D (R_{δ_d}, K_{δ_d}). In line 10, ζ_z obtains $K_{\delta_d \zeta_z}$ by using its secret key K_{ζ_z}. Then δ_d receives the message $M_{\zeta_z \delta_d}$ from ζ_z, and computes its own ID and

fog node ID ζ_zID by using $K_{\delta_d \zeta_z}$. If the session key between δ_d and ζ_z are not the same, the secure communication channel will not be built.

5.2.5 Scalability and compatibility of the system

The authentication protocol's scalability requirements are identified and defined, encompassing factors such as the number of users, concurrent authentication requests, network traffic volume, and system response time. The protocol is designed to leverage the capabilities of 5G networks for connecting entities. The inherent characteristics of 5G networks enable them to handle a substantial number of user requests efficiently. Consequently, as the user count increases, the 5G network is expected to maintain optimal request processing without experiencing delays. Moreover, it is important to note that the authentication process does not involve storing any authentication-related information by the 5G service provider. This implies that the system's efficiency will not be compromised by adding more edge users or fog nodes. These characteristics collectively indicate that the proposed protocol exhibits a high degree of scalability.

In the IoT environment, numerous users utilize a variety of devices that often originate from different manufacturers. In the designed protocol, both edge users and fog nodes access the 5G network through the eSIM card. This approach ensures that, despite the diversity of devices used by edge users, they can rely on the eSIM card for seamless data transmission. Consequently, this significantly mitigates compatibility conflicts, data synchronization issues, and the requirement for additional middleware components to facilitate communication between different tiers. By leveraging the standardized eSIM technology, the protocol promotes enhanced compatibility and streamlined communication within the IoT ecosystem.

Chapter 6

6 Performance evaluation

This chapter delves into a comprehensive analysis of the performance of the protocol that has been designed. The primary objective of this evaluation is to assess the protocol's efficiency across various metrics, including computational cost, signaling cost, communication cost, and storage cost. To achieve this, a comparative analysis is conducted against existing protocols to ascertain the protocol's superiority and identify areas for improvement.

First and foremost, the computational cost of the designed protocol is thoroughly examined. This encompasses an in-depth investigation of the computational resources required to execute the protocol's operations. By evaluating the time complexity, an understanding of the protocol's computational demands is gained.

Furthermore, the signaling cost of the protocol is carefully assessed. This entails analyzing the overhead incurred during the signaling process, such as message exchanges. The evaluation aims to quantify the efficiency of the protocol's signaling mechanisms, ensuring that they strike a balance between effectiveness and resource consumption.

In addition, the communication cost of the protocol is examined, especially the amount of data exchanged between network nodes during protocol execution. An assessment is made to determine the protocol's efficiency in utilizing network resources, with an emphasis on optimizing data transfer and minimizing unnecessary communication overhead.

Lastly, the storage cost of the protocol is also examined. This involves analyzing the storage requirements to support the protocol's operation. By evaluating the size of data, the evaluation seeks to optimize the protocol's storage efficiency and minimize resource consumption.

To provide a robust evaluation, a comparative analysis is performed against existing protocols. By benchmarking the designed protocol against well-established alternatives, a comprehensive understanding of its strengths and weaknesses is gained. This analysis enables the identification of areas where the protocol outperforms existing solutions and areas where further enhancements can be made.

For the evaluation of costs, we suppose that the length of random number is 128 bits (L_R), the AES encryption/decryption block size is 128bits (L_{AES}), the key length of AES is 128 bits (L_K), the key length of 5G-AKA/4G EPS-AKA is 256 bits(L_{AKA-K}), the length of hash function (SHA-256) is 256 bits (L_H), the identity, temporary identity and anonymous identity are length of 128 bits (L_{ID}), the length of sequence number SQN in 5G-AKA is 48 bits (L_{SQN}). The timestamp is 32 bits (L_{TS}). The length of symmetric polynomial is bits 384 (L_{SP}). NTRU encryption/decryption block size is 160 bits (L_{NTRU}).

In [89], identity (L_{ID*}) and timestamp (L_{TS*}) use 64 bits each. The elliptic curve point (L_{ECCp}) is 320 bits. Hash function (L_{H*}) is 160 bits. Symmetric key (L_{SyK}) is 128 bits for encryption.

In TLS 1.3, the master key size is 384 bits (L_{MK}), the pre-master key size is 256 bits (L_{PMK}), the client/server random number has a size of 256 bits ($L_{C/SR}$). During the ECDHE process, the client/server private key length is 256 bits ($L_{C/SPrK}$), and the public key length is 520 bits ($L_{C/SPuK}$). The session resumption key size is 256 bits (L_{SRk}). In general, the size of ClientHello or ServerHello message depends on how many extensions the message has, such as server name, session ticket and so on. The maximum size of a ClientHello message or ServerHello message is limited to 524280 bits. We suppose that the ClientHello message has one extension, and its size is 480 bits (L_{CH}), the ServerHello message also has one extension, and its size is 320 bits (L_{SH}). The finished message has a fixed length of 288 bits (L_F). Also, the certificate has a length of 512 bits (L_{Cert}), certificate verify length is 512 bits (L_{CertV}) and the signed certificate timestamp size is 64 bits (L_T).

6.1 Authentication between entities

6.1.1 Computational Cost

The comparison of our designed protocol with established standards such as 5G-AKA, 4G EPS-AKA, and TLS 1.3 in terms of computational costs is driven by the need to comprehensively evaluate the efficiency and feasibility of our innovation. By benchmarking against these well-established protocols, this chapter aim to ascertain the computational overhead incurred by our solution and its competitive edge in terms of resource utilization. This analysis serves to provide a robust understanding of the computational demands imposed by our protocol in comparison to its counterparts and shedding light on potential advantages.

In our proposed authentication protocol, only AES-128 encryption technology is used. In 5G-AKA protocol, the encryption technology used to encrypt user ID is Elliptic Curve Integrated Encryption Scheme (ECIES). In 4G EPS – AKA, it uses a Hash-based message authentication code key-derivation function (HKDF). In the transport layer security (TLS 1.3) protocol, we chose TLS_AES_128_GCM_SHA256 as the cipher suite. The CPU (Central Processing Units) running time simulation results are shown in Table 3. This simulation is done on Google Colab (Intel(R) Xeon(R) CPU @ 2.20GHz). Figures 7 to 10 show the authentication and key agreement phase of each protocol.

Table 3: The Running Time of each Operations

Operation	Time (ms)
Hash Function (T_H)	0.107
ECIES Encryption (T_{ECIES_E})	1.166
ECIES Decryption (T_{ECIES_D})	1.164
AES Encryption (T_{AES_E})	0.217

AES Decryption (T_{AES_D})	0.243
ECDHE processing (T_{ECDHE})	1.535
HKDF processing (T_{HKDF})	0.578
Verify certificate (T_{VerC})	0.138

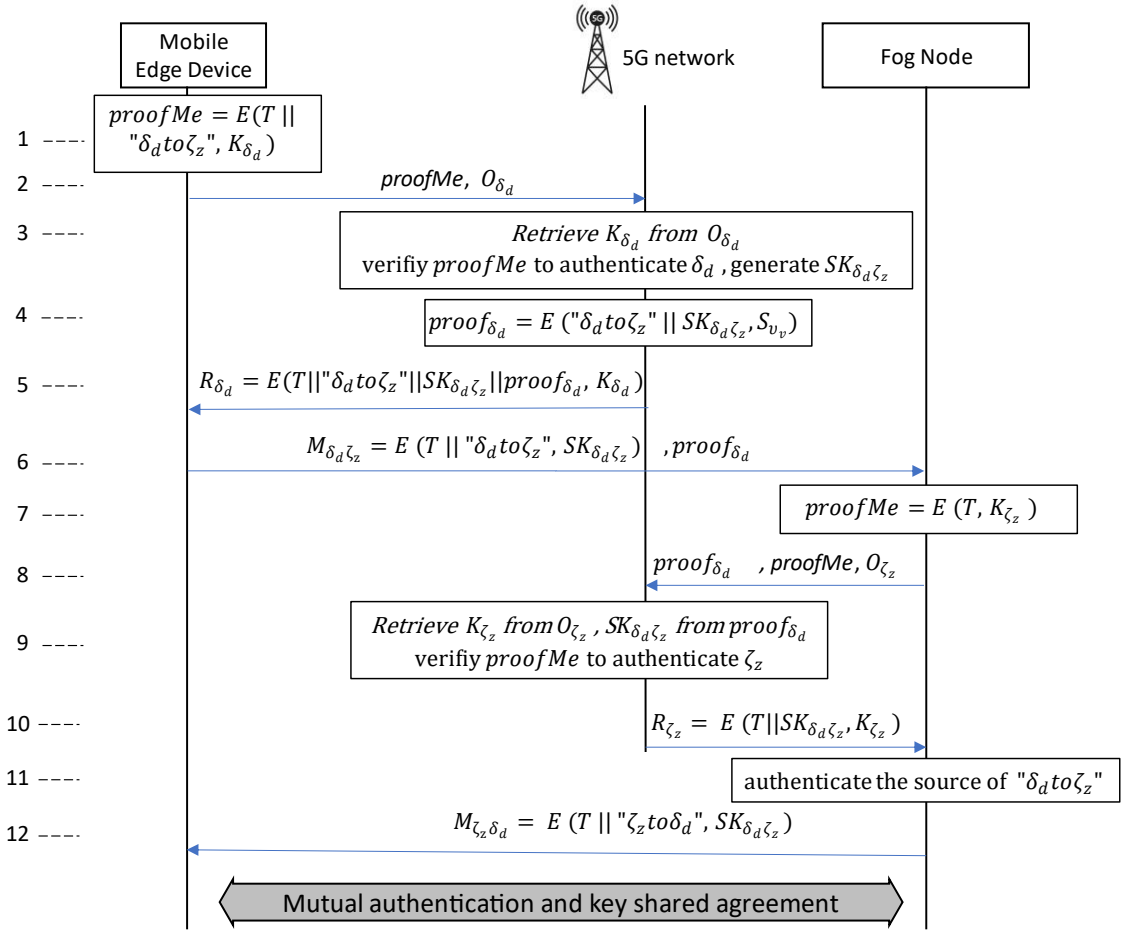

Figure 7: 3Tier-AKA authentication and key agreement procedure

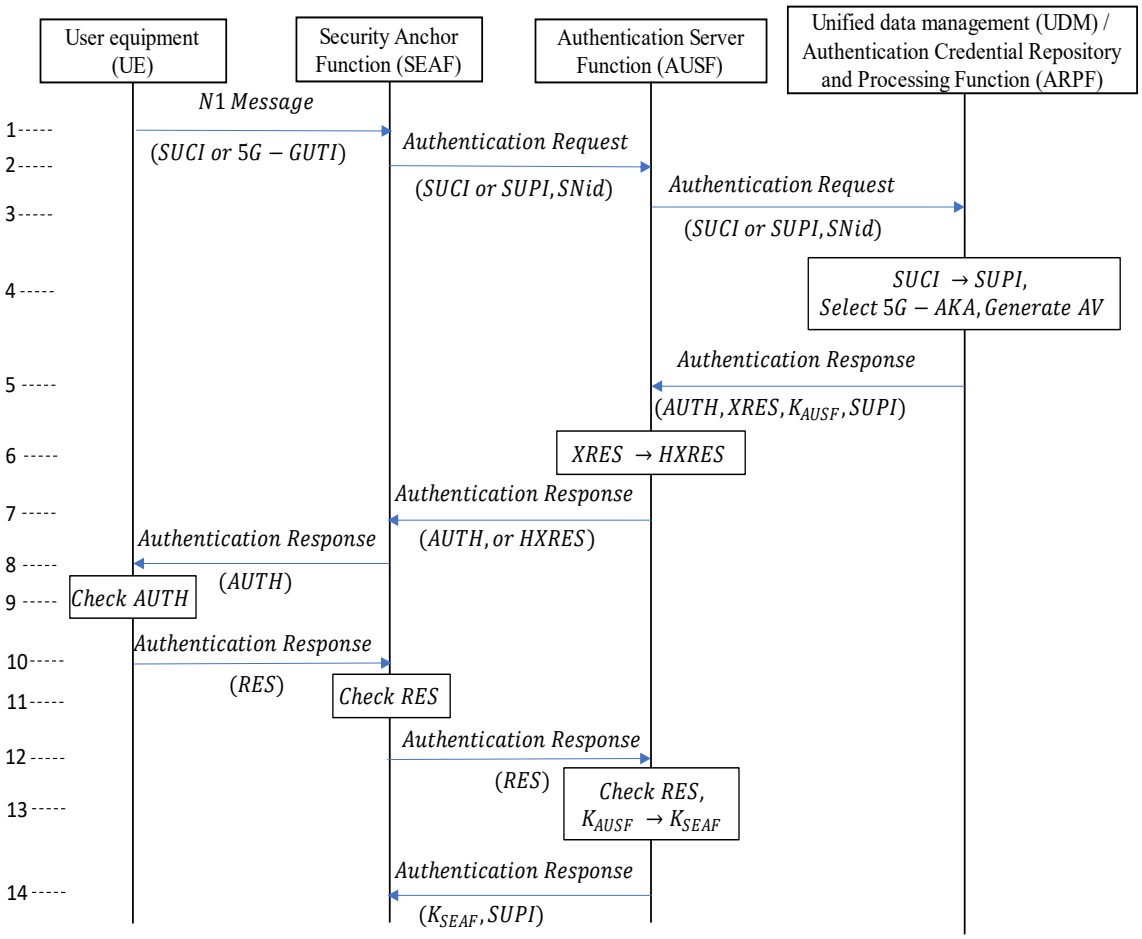

Figure 8: 5G – AKA [74]

Table 4 below shows abbreviations used in 5G – AKA protocol.

Table 4: Abbreviations in 5G – AKA

Notation	Description
SUCI	Subscription Concealed Identifier
GUTI	Globally Unique Temporary Identity
SUPI	Subscription Permanent Identifier
SNid	Serving network ID
AV	Authentication Vector
AUTH	Authentication token
RES	Response Token
XRES	Expected Response Token
HXRES	Hash of the expected response token
K_{AUSF}	Key used to derive other keys for authentication and encryption
K_{SEAF}	Anchor key (in 5G, for the Security Anchor Function)

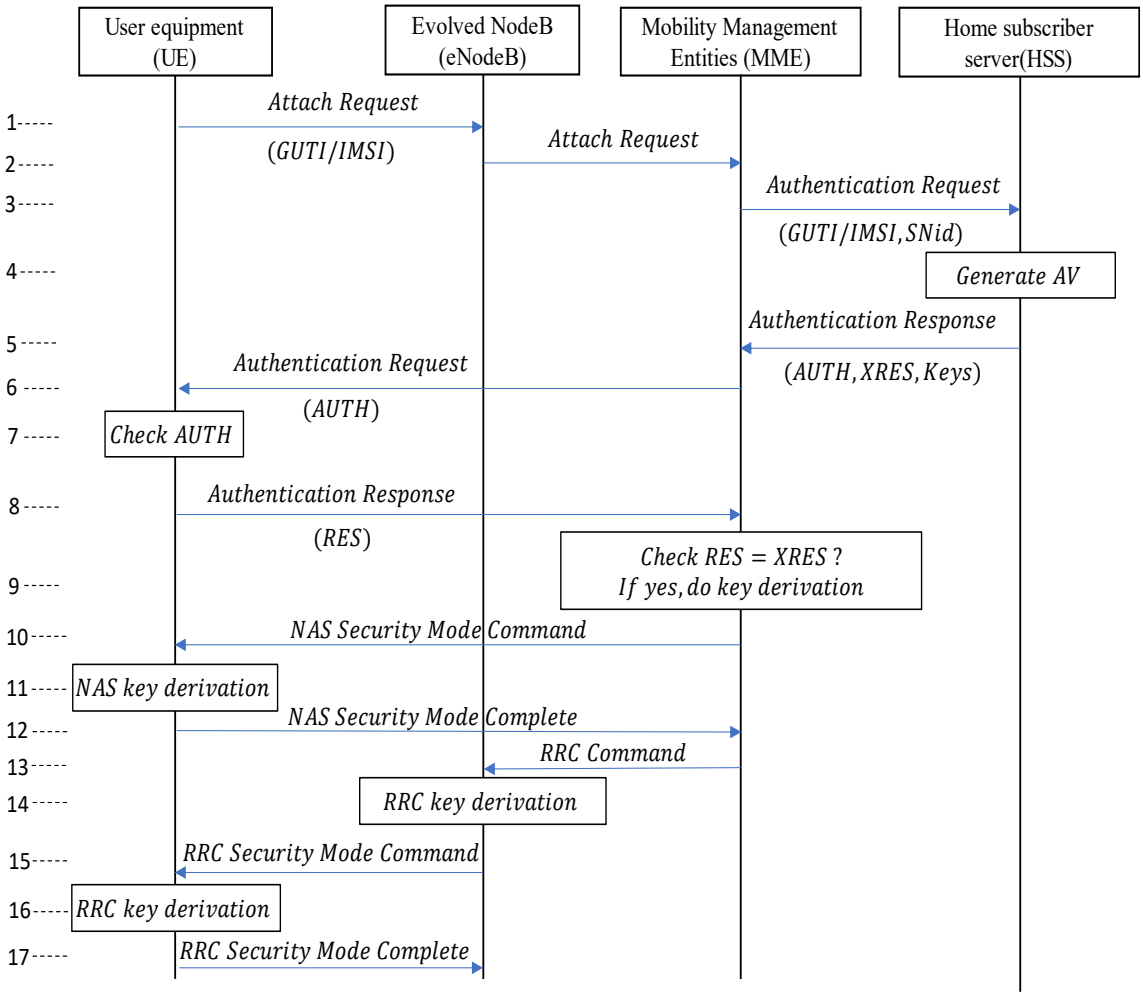

Figure 9: 4G EPS – AKA [53]

Table 5 below shows abbreviations used in 4G EPS – AKA protocol.

Table 5: Abbreviations in 4G EPS – AKA

Notation	Description
GUTI	Globally Unique Temporary Identity
IMSI	International mobile subscriber identity
XRES	Expected Response Token
AUTH	Authentication token
RRC	Radio Resource Control
SNid	Serving network ID
AV	Authentication Vector
RES	Response Token
NAS	Non-access stratum

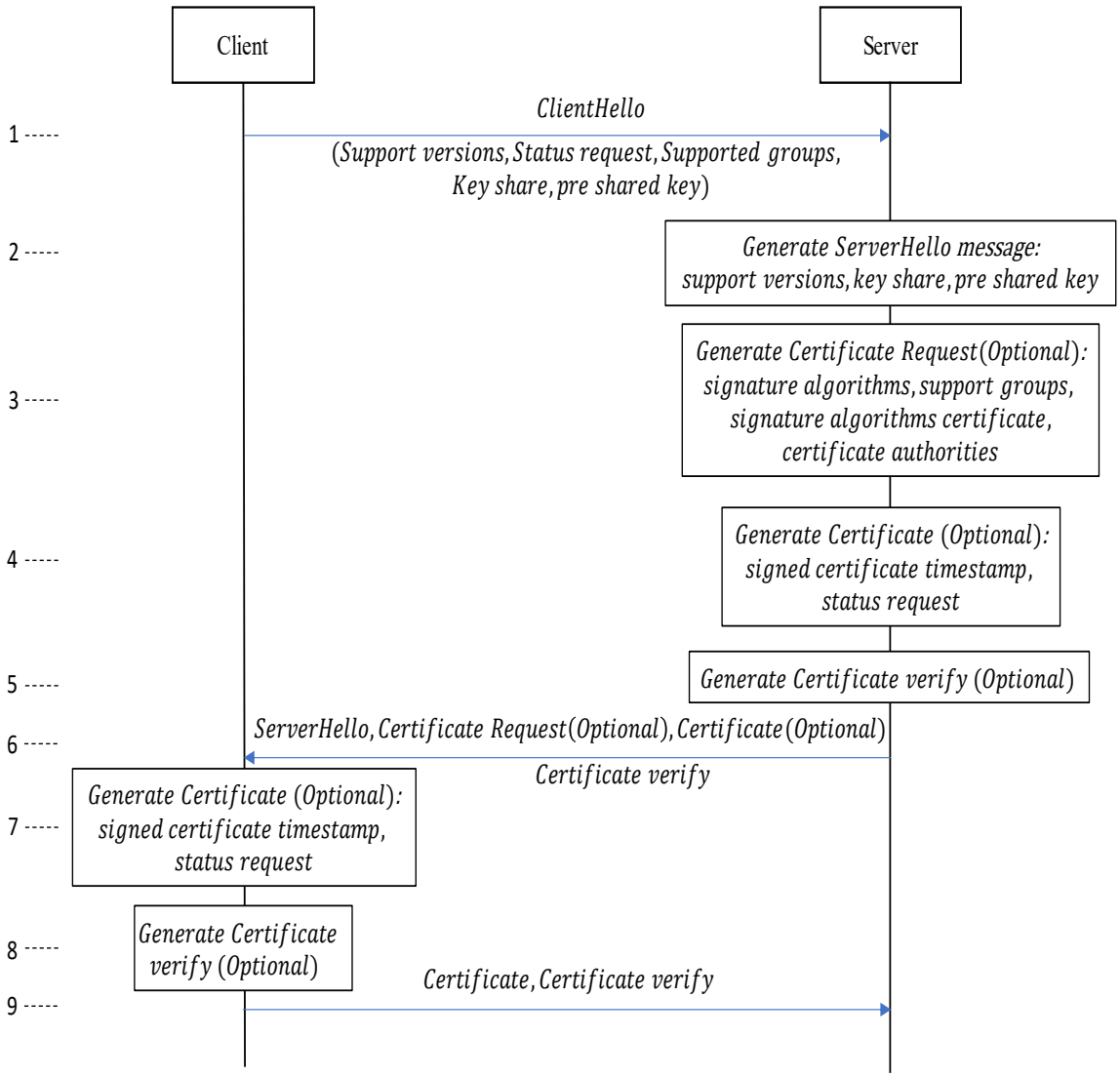

Figure 10: TLS 1.3 Handshake [80]

In 3Tier – AKA (Please see Figure 7), the edge user needs to compute two AES encryptions in line 1 and line 6 and perform two AES decryptions after line 5 and line 12. For the total computational cost, in the 5G network, it performs AES decryption in line 3 and line 9, and AES encryption in line 4, line 5, and line 10. In fog node, there are 2 AES decryptions in line 6 and line 10, and 2 encryptions in line 7 and line 12. Total computational cost is 7 AES encryptions and 8 AES decryptions.

In the 5G-AKA process (Please see Figure 8), the edge user first calculates its Subscription Concealed Identifier (SUCI) or Globally Unique Temporary Identity (GUTI) using Elliptic Curve Integrated Encryption Scheme (ECIES). It then transmits this identifier to the Security Anchor Function (SEAF) in line 1. Moving to line 2, SEAF forwards the authentication message, along with the serving network ID, to the Authentication Server Function (AUSF). In line 3, AUSF further sends the authentication message to either the Unified Data Management (UDM) or the Authentication Credential Repository and Processing Function (ARPF). Subsequently, in the next line, UDM/ARPF generates an authentication response message. Upon receiving the authentication response message after line 5, the edge user needs to compute the response, which involves performing 7 hash functions. This computation adds to the total computational cost. On the other hand, the 5G network decrypts the SUCI and calculates the authentication messages using 9 hash functions. The authentication process is in line 9, the edge user authenticates the 5G core network (SEAF, AUSF, UDM, ARPF). In lines 11 and 13, the SEAF and AUSF, respectively, authenticate the edge user, completing the mutual authentication between the entities involved in the 5G-AKA process.

In the 4G EPS-AKA (Evolved Packet System - Authentication and Key Agreement) (Please see Figure 9), the process shares similarities with 5G-AKA, but there is a notable difference in the initial steps. Unlike 5G-AKA, there is no ID encryption process at the beginning of 4G EPS-AKA.

In TLS 1.3 (Please see Figure 10), the authentication and key exchange process involves several steps. First, the client computes the shared pre-master key, and calculates the master key. These pieces of information are then forwarded to the server in line 2. Upon receiving the ClientHello message from the client, the server computes the shared pre-master key and uses the HKDF (HMAC-based Extract-and-Expand Key Derivation Function) to derive the master key. Lines 3, 4, and 5 represent optional steps for authentication, which may or may not be used depending on the specific configuration. Once the client receives the ServerHello message in line 6, it verifies the server's certificate and generates a certificate message (line 7). Subsequently, after line 9, the server verifies the client's certificate, completing the mutual authentication process. In

summary, TLS 1.3 facilitates secure communication between the client and server by exchanging cryptographic keys and verifying certificates to establish a trusted connection. The mutual authentication ensures that both parties can be confident in each other's identities during the communication session.

The computational costs of several protocols are shown in Table 6. The 4G EPS-AKA only costs 0.856 ms in edge tier. However, edge user IDs are not anonymized during the authentication process. Compared to other protocols, 3Tier – AKA has the highest efficiency.

Table 6: Computational Cost

Protocol	Edge User Computational Cost (ms)	Total Computational Cost (ms)
3Tier - AKA	$2T_{AES\_E} + 2T_{AES\_D} = 0.920$	$7T_{AES\_E} + 9T_{AES\_D} = 3.706$
5G-AKA [53]	$T_{ECIES\_E} + 7T_H = 1.915$	$T_{ECIES\_E} + T_{ECIES\_D} + 15T_H = 3.935$
4G EPS-AKA [53]	$8T_H = 0.856$	$16T_H = 1.712$
TLS 1.3 [80]	$T_{ECDHE} + T_{HKDF} + T_{VerC} = 2.251$	$2T_{ECDHE} + 2T_{HKDF} + 2T_{VerC} = 4.502$

6.1.2 Signaling Cost

In the context of a communication protocol involving an Edge device, a 5G service provider, and a Fog node, this chapter present a comparison of different signaling messages involved in various protocols. The primary focus is on the number of signaling messages exchanged.

The table provided (Table 7) summarizes the findings of this comparison. The proposed protocol involves six different types of signaling messages that are exchanged between the Edge device, the 5G service provider, and the Fog node. These signaling messages serve different purposes, including authentication request, response messages, and key agreement messages. The widely used TLS 1.3 protocol requires only three signaling messages for its operation. However, the size of each message in TLS 1.3 is comparatively larger than the messages used in the other protocols being considered. The 5G-AKA protocol involves nine signaling messages for its operation. This higher number of messages can result in increased communication overhead and potentially higher latency. Similarly, the 4G EPS-AKA protocol requires eight signaling messages. Again, this higher number of messages might lead to increased overhead. The protocol proposed in the paper, referred to as 3Tier-AKA, stands out in this comparison. It requires only six signaling messages for its operation, which is fewer than both 5G-AKA and 4G EPS-AKA. This reduced number of signaling messages in 3Tier-AKA makes it more efficient in terms of communication overhead and potentially contributes to lower latency compared to the other authentication mechanisms.

Table 7: Signaling Cost

Protocol	Signaling Cost
3Tier - AKA	6
5G-AKA [53]	9
4G EPS-AKA [53]	8
TLS 1.3 [80]	3

6.1.3 Communication Cost

3Tier – AKA: In lines 2, 5, 6, 8, 10, and 12, there are a total of 10 AES encrypted messages. Please see Figure 7.

5G – AKA: In line 1 and line 11, there are a total of 6 IDs involved during the communication process. During line 3, it includes a sequence number, a random number, two hash functions, and a key. Then in line 5, there is a random number and two hash functions transferred. In line 7 and line 9, there are total of four hash functions transferred. Please see Figure 8.

4G EPS – AKA: In the 4G EPS – AKA, it is similar to 5G AKA, except there is no ID encryption process in the beginning. Please see Figure 9.

TLS 1.3: In line 1, the client sends ClientHello message to the server. Once, server receives the message, it sends ServerHello message, certification, certification timestamp and certification verify back to client. After the client authenticates the server, in line 3, it transfers its certification, certification timestamp and certification verify back to server. Please see Figure 10.

The 3Tier – AKA only performs 10 AES encryption/decryption operations. Hence, the overall communication cost of 3Tier – AKA amounts to 1280 bits, significantly lower than that of other protocols.

Table 8 shows the total communication cost of our protocol and other related schemes during the authentication and key agreement phase.

Table 8: Total Communication Cost (bits)

Protocol	Total Communication Cost (bits)
3Tier - AKA	$10L_{AES} = 1280$
5G-AKA [53]	$6L_{ID} + L_{SQN} + L_{AKA-K} + 8L_H + 2L_R = 3376$
4G EPS-AKA [53]	$4L_{ID} + 2L_R + 2L_{SQN} + 7L_H + 2\ L_{AKA-K} = 3168$
TLS 1.3 [80]	$2L_T + 2L_{Cert} + 2L_{CertV} + L_{CH} + L_{SH} + L_F = 3264$

6.1.4 Storage Cost

The edge device storage cost after the authentication and key agreement phase are shown below in Table 9.

Table 9: Storage Cost (bits)

Protocol	Storage Cost (bits)	5G/4G network / Server (bits)
3Tier - AKA	$2L_{ID} + 2L_K + L_{AES} = 640$	$L_K = 128$
5G-AKA [53]	$2L_{ID} + 2L_{AKA-K} + L_{SQN} = 688$	$2L_{ID} + 2L_{AKA-K} + L_{SQN} = 688$
4G EPS-AKA [53]	$2L_{ID} + 3L_{AKA-K} + L_{SQN} = 816$	$2L_{ID} + 3L_{AKA-K} + L_{SQN} = 816$
TLS 1.3 [80]	$L_T + L_{Cert} + L_{CertV} + L_{SRk} + L_{C/SR} + L_{MK} + L_{C/SPrK} + L_{C/SPuK} = 3033$	$L_T + L_{Cert} + L_{CertV} + L_{SRk} + L_{C/SR} + L_{MK} + L_{C/SPrK} + L_{C/SPuK} = 3033$

In our protocol, the edge device only needs to store its identity δ_dID, a Token O_{δ_d}, a secret key K_{δ_d} and a session key $K_{\delta_d\zeta_z}$. Fewer storage requirements mean more edge users can be accommodated. The edge user storage cost of 3Tier – AKA is 640 bits, which indicates that this protocol is more lightweight than the other three protocols.

6.2 Handover Authentication

6.2.1 Computational Cost

In our proposed authentication protocol, only AES-128 encryption technology is used. In FogHA [57], the encryption technology used is t-degree symmetric polynomial based. Based on references [90] and [91], the computation time of a symmetric polynomial message authentication code (T_{MAC}) closely aligns with the running time of a cryptographic hash function, particularly when the entity's identity consists of 128 bits. Specifically, when polynomial t degree is 100, calculating a t-degree symmetric polynomial (T_{SP}) takes approximately 16 times longer than computing T_{MAC} [57]. The Number Theory Research Unit (NTRU) encryption is approximately 20 times slower than an AES implementation [92]. The symmetric encryption/decryption key used in [89] is 128 bits. Then the symmetric encryption/decryption operation time will be the same as AES encryption/decryption. The CPU (Central Processing Units) running time simulation results are shown in Table 10. This simulation is done on Google Colab (Intel(R) Xeon(R) CPU @ 2.20GHz). Figures 11 to 14 show the authentication and key agreement phase of each protocol.

Table 10: Computational Cost (handover)

Operation	Time (ms)
Hash Function (T_H)	0.107
AES Encryption (T_{AES_E})	0.217
AES Decryption (T_{AES_D})	0.243
T-degree symmetric polynomial (T_{SP})	1.712
NTRU Encryption (T_{NTRU_E})	4.340
NTRU Decryption (T_{NTRU_D})	4.860
Elliptic curve point multiplication operation (T_{ECC})	5.227
Symmetric encryption operation(T_{SE_E})	0.217
Symmetric decryption operation(T_{SE_D})	0.243

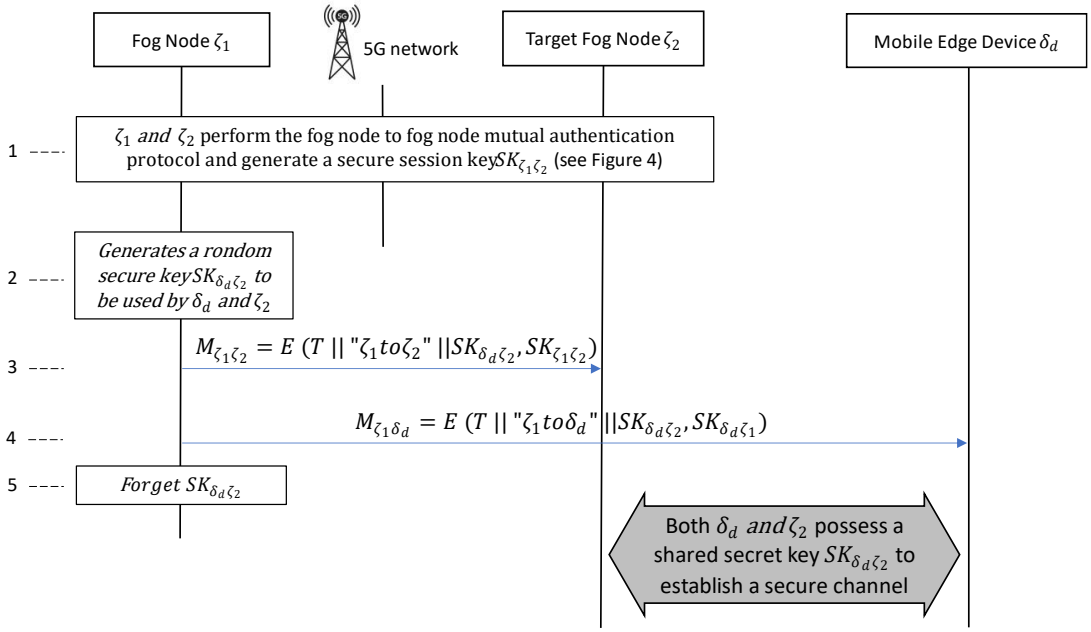

Figure 11: 3Tier-AKA handover authentication procedure

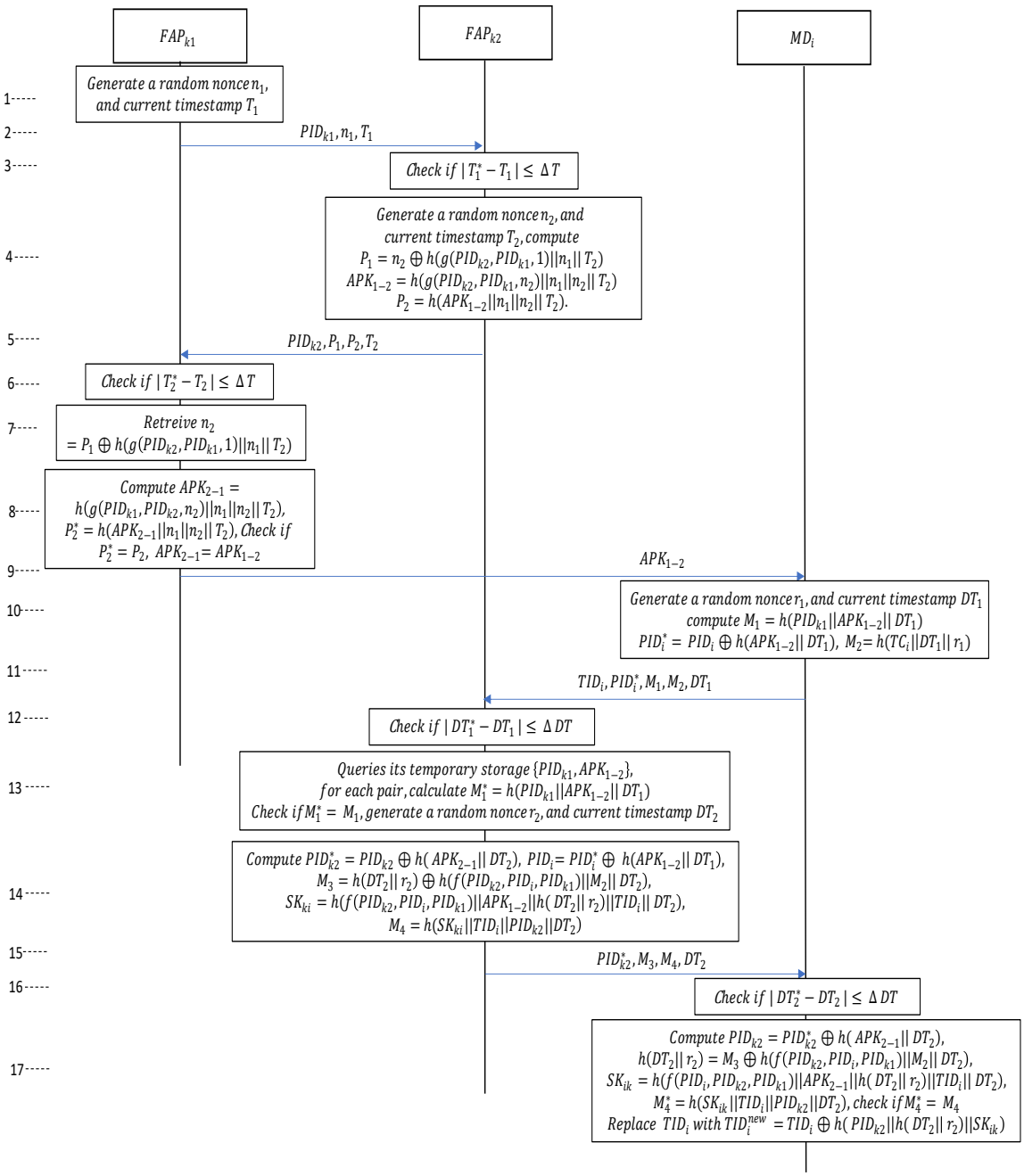

Figure 12: FogHA handover authentication procedure[57]

Table 11 below shows abbreviations used in FogHA protocol.

Table 11: Abbreviations in FogHA

Notation	Description
FAP_k	k^{th} fog access point
MD_i	Mobile devices of user i
PID	Pseudo identities
APK	Pre-negotiation temporary key
Δ T	Maximum transmission delay
TC	Credentials of user
TID	Temporary identity
SK	Session key

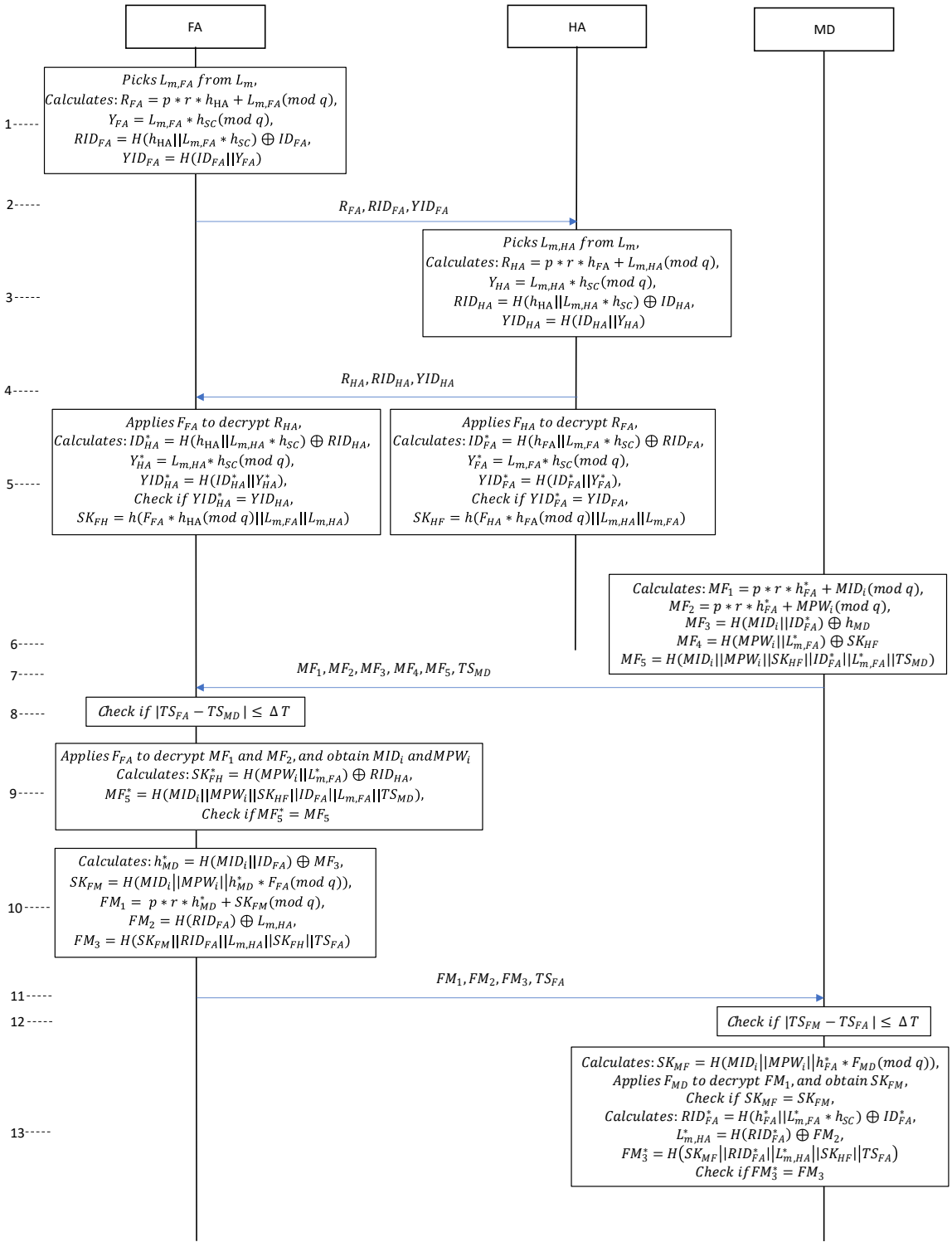

Figure 13: Quantum-resistant handover authentication protocol procedure [93]

Table 12 below shows abbreviations used in Quantum-resistant handover authentication protocol.

Table 12: Abbreviations in Quantum-resistant handover authentication protocol

Notation	Description
FA	Foreign agent
HA	Home agent
MD	Mobile device
L_m, $L_{m,HA}$, $L_{m,FA}$, p, r	Polynomial
h, F	Public/private key of entity
h_{SC}	trusted third system center public key
TS	Timestamp
SK	Session key
MID_i, MPW_i	Hashed ID and password

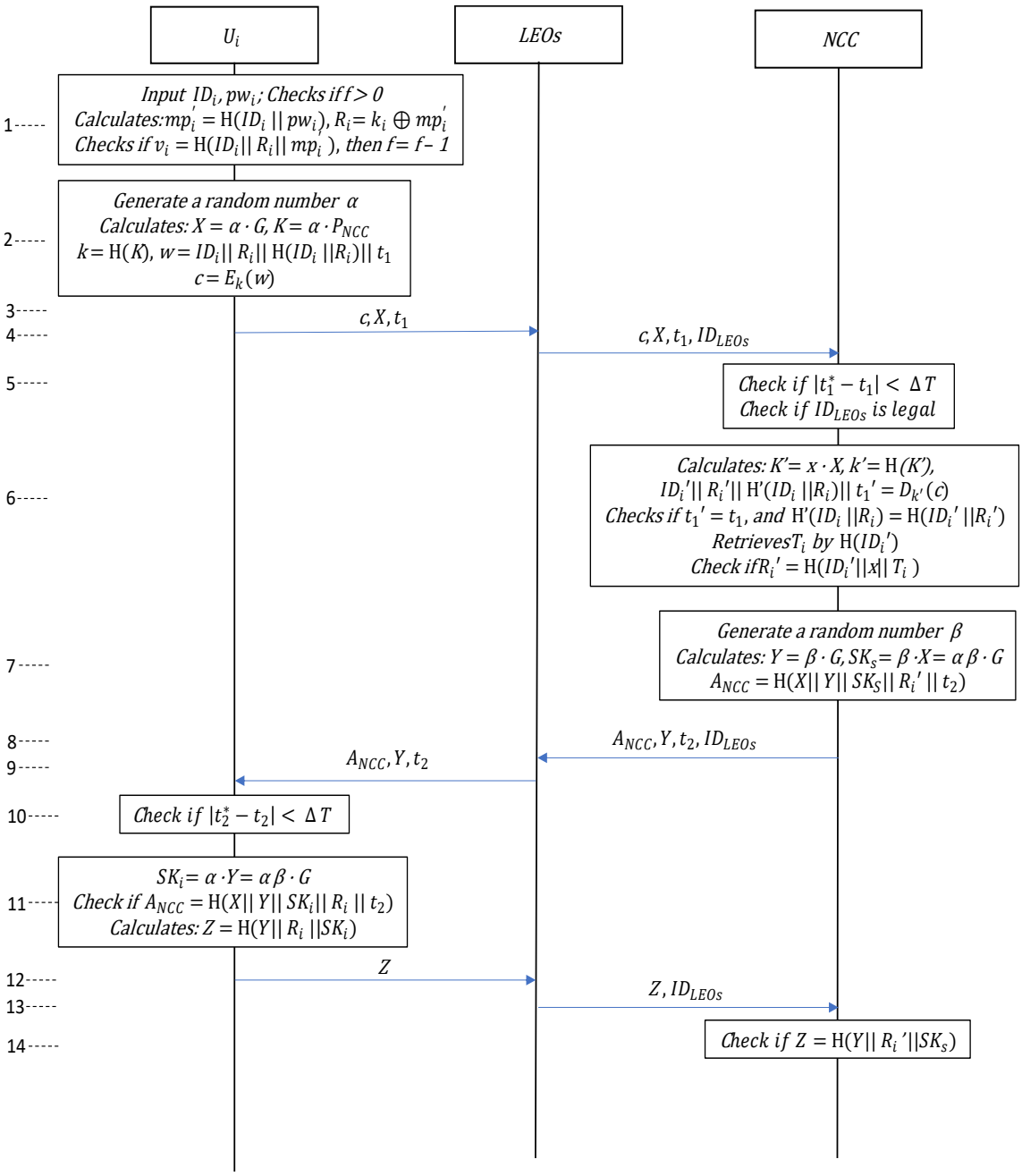

Figure 14: Liu et al's scheme authentication protocol procedure [89]

Table 13 below shows abbreviations used in Liu et al's scheme.

Table 13: Abbreviations in Liu et al's scheme

Notation	Description
U_i	ith user
LEOs	Low-earth-orbit satellite
NCC	Network control center
ID, pw	Entity's identity, password
v_i, k_i, f	parameters stored in a smart card
G	A base point over Ep(a, b) with prime order n
P_{NCC}	The public key of NCC
x	The private key of NCC
SK	Session key
t	Timestamp
t’	current time
T_i	User i registration time

In 3Tier – AKA (Please see Figure 11), the mutual authentication among fog nodes has been previously executed, hence this aspect of the cost is not encompassed within the handover authentication section. The edge user is required to carry out one AES decryption after line 4. As for the old fog node, it necessitates two AES encryptions in lines 3 and 4. In the target fog node, it performs one AES decryption after line 3. The overall computational cost encompasses 2 AES encryptions and 2 AES decryptions.

In the FogHA protocol (Please see Figure 12), the handover process begins with the first fog access point (FAPk1) generating a timestamp and a random number in line 1. Subsequently, FAPk1 proceeds to authenticate FAPk2 in line 8. During this exchange, a temporary session key is established between the FAPs, denoted in lines 4 and 8. Once the temporary session key is in place, FAPk1 sends it to the mobile device in line 9. The mobile device then computes the authentication message and forwards it to the second fog access point (FAPk2) in line 11. In line 13, FAPk2 performs the authentication of the mobile device. As part of this process, FAPk2 generates the FAP-MD session key, as described in line 14. With FAPk2's authentication successful, the mobile device verifies FAPk2 and proceeds to calculate the session key in line 17, finalizing the handover procedure in the FogHA protocol. The mobile device of the edge user needs to execute 8 hash functions and 1 t-degree symmetric polynomial computation in line 10 and line 17. As for the first fog access point, it performs 3 hash functions and 2 t-degree symmetric polynomial computations in line 7 and line 8. Meanwhile, the second fog access point involves line 4 in the handover pre-negotiation process and line 13 and line 14 in the handover process, requiring 10 hash functions and 3 t-degree symmetric polynomial computations. Overall, this protocol necessitates a total of 21 hash functions and 6 t-degree symmetric polynomial executions.

In the quantum-resistant handover authentication protocol (Please see Figure 13), the process unfolds as follows: During line 1 and 2, the foreign agent (FA) computes the authentication message and sends it to the home agent (HA). Subsequently, in line 3, HA generates its authentication message and transmits it back to FA. In line 5, both HA and FA mutually authenticate each other, leading to the generation of a temporary session key shared between them. Moving to line 6, the mobile device (MD) calculates its

authentication message and transmits it to FA for verification. Upon successful authentication of MD by FA in line 9, FA proceeds to generate a new session key. It then securely forwards the encrypted session key and relevant parameters to MD. In line 13, MD performs verification of FA's identity and calculates the new session key, effectively completing the quantum-resistant handover authentication protocol. The mobile device performs 2 NTRU (Number Theory Research Unit) encryptions and 1 NTRU decryption in line 6 and line 13, respectively. Additionally, it involves 7 hash functions. Considering the overall computational cost, there are 5 NTRU encryptions, 5 NTRU decryptions, and 23 hash functions in total.

In Liu et al.'s scheme (Please see Figure 14), the user initiates the process by generating a timestamp, elliptic curve point parameters, and encrypted ID. These are then transmitted to Low-earth-orbit satellites (LEOs) in line 3. Subsequently, LEOs forward the message and its ID to the Network Control Center (NCC) in the following line. Upon receiving the message, NCC first verifies the identity of LEOs in line 5. In line 6, NCC proceeds to verify the user's identity. Upon successful verification, NCC calculates the session key and the response message in line 7. Once the user receives the response message, it performs a verification of NCC's identity in line 11. Subsequently, in line 11, the user generates its session key. To ensure the session keys held by both the user and NCC are identical, the message Z is transmitted to NCC in lines 12 and 13. NCC then checks the session key in line 14, ensuring consistency and completing the authentication and session key establishment process. The lines 1 and 2 involve the user inserting their smart card into a card reader, where they perform two elliptic curve point multiplication operations, four hash function operations, and one symmetric encryption operation. In line 11, the user executes one elliptic curve point multiplication operation and two hash function operations. As for the network control center (NCC), lines 6 and 7 require one symmetric decryption operation, five hash function operations, and three elliptic curve point multiplication operations. In line 14, the NCC performs one hash function operation.

The computational costs of several protocols are shown in Table 14. The Liu et al.'s scheme has the highest costs in edge tier. Compared to other protocols, 3Tier – AKA has the highest efficiency.

Table 14: Computational Cost (Handover)

Protocol	Edge User Computational Cost (ms)	Total Computational Cost (ms)
3Tier - AKA	T_{AES_D} = 0.243	$2T_{AES\_E}$ + $2T_{AES\_D}$ = 0.920
FogHA [57]	$8T_H$ + T_{SP} = 2.568	$21T_H$ + $6T_{SP}$ = 12.519
Quantum-resistant handover authentication protocol [93]	$2T_{NTRU\_E}$ + T_{NTRU_D}+ $7T_H$ = 14.289	$5T_{NTRU\_E}$ + $5T_{NTRU\_D}$+ $23T_H$ = 48.461
Liu et al's scheme [89]	$3T_{ECC}$ + T_{SE_E} + $6T_H$ = 16.540	$6T_{ECC}$ + T_{SE_E} + T_{SE_D} + $12T_H$ = 33.106

6.2.2 Signaling Cost

In the context of a communication protocol involving an Edge device, a 5G service provider, and a Fog node, this chapter present a comparison of different signaling messages exchanged among these entities. The focus is on evaluating the efficiency and effectiveness of various protocols in terms of the number of signaling messages and overall performance.

The findings are presented in Table 15, which outlines the characteristics of different protocols. The proposed protocol stands out for its efficiency. It involves only two signaling messages that are exchanged between the Edge device, 5G service provider, and Fog node. These two messages serve the purposes of handover and key agreement. In contrast, another protocol proposed by Liu et al. is highlighted as having the highest signaling cost, involving six signaling messages. This comparison suggests that Liu et al.'s scheme might result in higher communication overhead and potentially increased

latency. Comparing the proposed protocol with two other schemes, namely FogHA and Quantum-resistant handover authentication protocols, designed protocol has lower signaling costs. Specifically, FogHA involves five signaling messages, and the Quantum-resistant protocol involves four. In this regard, the proposed protocol stands out as having the smallest signaling costs among the three. Consequently, the proposed 3Tier - AKA protocol is the optimal choice among the evaluated options. Its key advantages lie in its minimal signaling costs when compared to other protocols.

Table 15: Signaling Cost (Handover)

Protocol	Signaling Cost
3Tier - AKA	2
FogHA [57]	5
Quantum-resistant handover authentication protocol [93]	4
Liu et al's scheme [89]	6

6.2.3 Communication Cost

3Tier – AKA: In lines 3 and 4, there are a total of 2 AES encrypted messages are transmitted. Please see Figure 11.

FogHA: The first fog access point initiates handover authentication communication by sending pre-negotiation information to the second fog access point in line 2. Once this exchange is completed, the first fog access point proceeds to transmit the pre-negotiation temporary key to the mobile device after line 5. Subsequently, in line 11 and line 15, authentication messages are transmitted between the second fog access point and the mobile device. The total communication cost are consisting of 5 IDs, 7 hash functions, 1 random number and 4 timestamps. Please see Figure 12.

Quantum-resistant handover authentication protocol: The messages exchanged between entities primarily consist of hash function operations and NTRU encrypted messages. In lines 2, 4, 7, and 11, there are a total communication cost of 5 NTRU encrypted messages, 9 hash function operations, and 2 timestamps. Please see Figure 13.

In Liu et al.'s scheme, the authentication message is initially sent from the user to the low-earth-orbit satellite (LEOs). Subsequently, LEOs appends its ID to the message and forwards it to the network control center (NCC). In lines 8 and 9, the response message is sent back to the user. Lines 12 and 13 involve the session key agreement process. The protocol includes two symmetric encryption operations, four elliptic curve point multiplication operations, four timestamps, three IDs, and four hash function operations transmitted in total. Please see Figure 14.

The 3Tier – AKA only performs 2 AES encryption operations. Hence, the overall communication cost of 3Tier – AKA amounts to 256 bits, significantly lower than that of other protocols. Table 16 shows the total communication cost of our protocol and other related schemes during the handover authentication phase.

Table 16: Total Communication Cost (bits) (handover)

Protocol	**Total Communication Cost (bits)**
3Tier - AKA	$2L_{AES} = 256$
FogHA [57]	$5L_{ID} + 7L_H + L_R + 4L_{TS} = 2688$
Quantum-resistant handover authentication protocol [93]	$5L_{NTRU} + 9L_H + 2L_{TS} = 3168$
Liu et al's scheme [89]	$3L_{ID^*} + 4L_{H^*} + 4L_{ECCp} + 2L_{SyK} + 4L_{TS^*} = 2624$

6.2.4 Storage Cost

The edge device storage cost after the handover phase is shown below in Table 17.

Table 17: Storage Cost (bits) (handover)

Protocol	**Storage Cost (bits)**	**5G/4G network / Server (bits)**
3Tier - AKA	$2L_{ID} + 2L_K + L_{AES} = 640$	$L_K = 128$
FogHA [57]	$4L_{ID} + 6L_H + 3L_{SP} = 3200$	$2L_{ID} + 2L_{SP} + 2L_H = 1536$
Quantum-resistant handover authentication protocol [93]	$3L_{ID} + 4L_H + 2L_R + 3L_{NTRU} = 1952$	$L_{ID} + 4L_{NTRU} + 4L_H = 1792$
Liu et al's scheme [89]	$L_{H^*} + L_K + 2L_{ID^*} + L_{ECCp} = 736$	$L_{TS^*} + 2L_{H^*} + 3L_K + 3L_{ID^*} + L_{ECCp} = 1280$

In our protocol, the edge device only needs to storage its identity δ_dID, fog node identity ζ_2ID, a Token O_{δ_d}, a secret key K_{δ_d} and a session key $K_{\delta_d\zeta_2}$. Less storage requirements means more edge users can be accommodated. The edge user storage cost of 3Tier – AKA is 640 bits, which indicates that this protocol is more lightweight than the other three protocols.

To summarize, this chapter provides a thorough and all-encompassing evaluation of the performance of the designed protocol. By conducting a meticulous analysis of the computational cost, signaling cost, communication cost, and storage cost, in comparison to existing protocols, a comprehensive assessment is achieved. The results of this evaluation demonstrate the superiority of our designed protocol in terms of performance, making it highly suitable for fog computing in a three-tier environment.

Chapter 7

7 Conclusion and future work

This chapter encompasses both the research conclusions and future works, presenting proposals to explore novel, distinct methodologies.

7.1 Conclusion

In this thesis, we have explored the development and implementation of a lightweight mutual authentication mechanism with handover function in the cloud-fog-edge environment, with a specific focus on integrating 5G communication technology. Our research aimed to address the crucial challenges of securing communication between edge devices, fog nodes, and cloud services while accommodating the dynamic mobility of edge devices within the distributed architecture and leveraging the capabilities of 5G technology. Also, handovers involve switching a device's connection from one access point to another. The authentication process during handovers introduces latency, which can affect real-time applications and services.

The proposed mutual authentication mechanism successfully achieved secure and efficient communication within the cloud-fog-edge environment. Throughout this research, we proposed and designed an efficient mutual authentication protocol that verifies the identities and access permissions of all entities involved in the communication process. By leveraging AES cryptographic techniques and secure handover mechanisms, we successfully facilitated seamless transitions between fog nodes during edge device mobility, ensuring uninterrupted access to services and resources.

Moreover, our research contributed to enhancing security in the cloud-fog-edge environment, mitigating potential risks of impersonation attacks, eavesdropping, and unauthorized access. The incorporation of robust security protocols in our authentication mechanism provided a shield against potential threats, safeguarding sensitive data and ensuring privacy during communication.

The effectiveness and performance of the lightweight mutual authentication mechanism were validated through comprehensive evaluations. A comparative analysis was conducted, contrasting the proposed protocol with existing ones, including TLS 1.3, 5G-AKA, and various handover protocols. The results showcased notable benefits, including reduced computational overhead, minimal communication latency, and seamless compatibility with resource-constrained edge devices.

Overall, this research contributes to enhancing security and reliability in the cloud-fog-edge environment. The lightweight mutual authentication mechanism presented in this thesis can serve as a foundational step toward ensuring seamless and secure communication between diverse entities within the distributed architecture. By capitalizing on the benefits of 5G technology, this research contributes to advancing secure and efficient communication in the context of emerging applications and services that rely on cloud-fog-edge systems empowered by 5G communication.

7.2 Future Work

The primary objective of this research is to tackle the emerging challenges associated with securing communication between edge devices, fog nodes, and cloud services, all while accommodating the mobility of edge devices. Further advancing the protocols introduced in this thesis represents a promising and valuable direction for future research endeavors.

Real-World Deployment and Evaluation: A practical implementation of the authentication mechanism will be carried out in a simulated cloud-fog-edge environment, and its performance will be thoroughly evaluated. The evaluation will include metrics such as authentication speed, handover efficiency, resource utilization, and overall system responsiveness.

Usability and User Experience: Human factors will also be taken into consideration during the development of the authentication mechanism. Usability studies and user feedback will be collected to ensure that the proposed solution is user-friendly and does not introduce unnecessary complexity.

Energy Efficiency: The future work will investigate energy-efficient authentication strategies for resource-constrained edge devices, aiming to prolong the battery life and improve overall energy efficiency in the cloud-fog-edge environment.

The ultimate goal of this future work is to contribute to the advancement of secure and seamless communication in the cloud-fog-edge environment, enabling a wide range of applications in areas such as Internet of Things (IoT), smart cities, and industrial automation. By addressing the challenges related to authentication and handover, this research will pave the way for more reliable and scalable cloud-fog-edge systems in the era of pervasive computing.

References

[1] S. Furnell, “Why users cannot use security,” *Comput. Secur.*, vol. 24, no. 4, pp. 274–279, Jun. 2005, doi: 10.1016/j.cose.2005.04.003.

[2] D. Zissis and D. Lekkas, “Addressing cloud computing security issues,” *Future Gener. Comput. Syst.*, vol. 28, no. 3, pp. 583–592, Mar. 2012, doi: 10.1016/j.future.2010.12.006.

[3] M. A. Bouras, B. Xia, A. O. Abuassba, H. Ning, and Q. Lu, “IoT-CCAC: a blockchain-based consortium capability access control approach for IoT,” *PeerJ Comput. Sci.*, vol. 7, p. e455, Apr. 2021, doi: 10.7717/peerj-cs.455.

[4] W. Voorsluys, J. Broberg, and R. Buyya, “Introduction to Cloud Computing,” in *Cloud Computing*, John Wiley & Sons, Ltd, 2011, pp. 1–41. doi: 10.1002/9780470940105.ch1.

[5] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, “Fog computing and its role in the internet of things,” in *Proceedings of the first edition of the MCC workshop on Mobile cloud computing*, in MCC ’12. New York, NY, USA: Association for Computing Machinery, Aug. 2012, pp. 13–16. doi: 10.1145/2342509.2342513.

[6] J. Gonzalez *et al.*, “Edge computing architecture and use cases,” *IBM Developer*. https://developer.ibm.com/articles/edge-computing-architecture-and-use-cases/ (accessed Aug. 03, 2023).

[7] A. Ouda, “A framework for next generation user authentication,” in *2016 3rd MEC International Conference on Big Data and Smart City (ICBDSC)*, Mar. 2016, pp. 1–4. doi: 10.1109/ICBDSC.2016.7460349.

[8] R. Shirey, *RFC2828: Internet Security Glossary*. USA: RFC Editor, 2000.

[9] A. Ometov, O. L. Molua, M. Komarov, and J. Nurmi, “A Survey of Security in Cloud, Edge, and Fog Computing,” *Sensors*, vol. 22, no. 3, Art. no. 3, Jan. 2022, doi: 10.3390/s22030927.

[10] “Adoption of cloud computing as innovation in the organization - Lewis Golightly, Victor Chang, Qianwen Ariel Xu, Xianghua Gao, Ben SC Liu, 2022.” https://journals.sagepub.com/doi/10.1177/18479790221093992 (accessed Jun. 22, 2023).

[11] L. Alhenaki, A. Alwatban, B. Alamri, and N. Alarifi, “A Survey on the Security of Cloud Computing,” in *2019 2nd International Conference on Computer Applications & Information Security (ICCAIS)*, May 2019, pp. 1–7. doi: 10.1109/CAIS.2019.8769497.

[12] H. Tabrizchi and M. Kuchaki Rafsanjani, “A survey on security challenges in cloud computing: issues, threats, and solutions,” *J. Supercomput.*, vol. 76, no. 12, pp. 9493–9532, Dec. 2020, doi: 10.1007/s11227-020-03213-1.

[13] S. Srivastava and S. P. Singh, “A Survey on Latency Reduction Approaches for Performance Optimization in Cloud Computing,” in *2016 Second International Conference on Computational Intelligence & Communication Technology (CICT)*, Feb. 2016, pp. 111–115. doi: 10.1109/CICT.2016.30.

[14] I. Stojmenovic and S. Wen, “The Fog computing paradigm: Scenarios and security issues,” in *2014 Federated Conference on Computer Science and Information Systems*, Sep. 2014, pp. 1–8. doi: 10.15439/2014F503.

[15] B. Martini and K.-K. R. Choo, “An integrated conceptual digital forensic framework for cloud computing,” *Digit. Investig.*, vol. 9, no. 2, pp. 71–80, Nov. 2012, doi: 10.1016/j.diin.2012.07.001.

[16] O. Osanaiye, S. Chen, Z. Yan, R. Lu, K.-K. R. Choo, and M. Dlodlo, “From Cloud to Fog Computing: A Review and a Conceptual Live VM Migration Framework,” *IEEE Access*, vol. 5, pp. 8284–8300, 2017, doi: 10.1109/ACCESS.2017.2692960.

[17] S. Yi, Z. Hao, Z. Qin, and Q. Li, “Fog Computing: Platform and Applications,” in *2015 Third IEEE Workshop on Hot Topics in Web Systems and Technologies (HotWeb)*, Nov. 2015, pp. 73–78. doi: 10.1109/HotWeb.2015.22.

[18] Y. Meng, M. A. Naeem, A. O. Almagrabi, R. Ali, and H. S. Kim, “Advancing the State of the Fog Computing to Enable 5G Network Technologies,” *Sensors*, vol. 20, no. 6, Art. no. 6, Jan. 2020, doi: 10.3390/s20061754.

[19] A. Baktayan, M. NA, and S. Alhomdy, “Fog Computing for Network Slicing in 5G Networks: An Overview,” vol. 07, Dec. 2018, doi: 10.4172/2167-0919.1000172.

[20] R. Das and M. M. Inuwa, “A review on fog computing: Issues, characteristics, challenges, and potential applications,” *Telemat. Inform. Rep.*, vol. 10, p. 100049, Jun. 2023, doi: 10.1016/j.teler.2023.100049.

[21] C. Gong, J. Liu, Q. Zhang, H. Chen, and Z. Gong, “The Characteristics of Cloud Computing,” in *2010 39th International Conference on Parallel Processing Workshops*, Sep. 2010, pp. 275–279. doi: 10.1109/ICPPW.2010.45.

[22] P. Hu, S. Dhelim, H. Ning, and T. Qiu, “Survey on fog computing: architecture, key technologies, applications and open issues,” *J. Netw. Comput. Appl.*, vol. 98, pp. 27–42, Nov. 2017, doi: 10.1016/j.jnca.2017.09.002.

[23] J. Montgomery and S. Lelli, “What is a Cloud SLA (Cloud Service-Level Agreement)?,” *Storage*, Jan. 2021.

https://www.techtarget.com/searchstorage/definition/cloud-storage-SLA (accessed Aug. 03, 2023).

[24] IBM, “What is Cloud Security? Cloud Security Defined | IBM.” https://www.ibm.com/topics/cloud-security (accessed Jun. 22, 2023).

[25] S. Yi, Z. Qin, and Q. Li, “Security and Privacy Issues of Fog Computing: A Survey,” in *Wireless Algorithms, Systems, and Applications*, K. Xu and H. Zhu, Eds., in Lecture Notes in Computer Science. Cham: Springer International Publishing, 2015, pp. 685–695. doi: 10.1007/978-3-319-21837-3_67.

[26] I. Ali, S. Sabir, and Z. Ullah, “Internet of Things Security, Device Authentication and Access Control: A Review.” arXiv, Apr. 20, 2022. doi: 10.48550/arXiv.1901.07309.

[27] J. Ren, D. Zhang, S. He, Y. Zhang, and T. Li, “A Survey on End-Edge-Cloud Orchestrated Network Computing Paradigms: Transparent Computing, Mobile Edge Computing, Fog Computing, and Cloudlet: ACM Computing Surveys: Vol 52, No 6,” Oct. 16, 2019. https://doi.org/10.1145/3362031 (accessed Jun. 23, 2023).

[28] A. Jangra and N. Mangla, “An efficient load balancing framework for deploying resource schedulingin cloud based communication in healthcare,” *Meas. Sens.*, vol. 25, p. 100584, Feb. 2023, doi: 10.1016/j.measen.2022.100584.

[29] M. Noaman, M. S. Khan, M. F. Abrar, S. Ali, A. Alvi, and M. A. Saleem, “Challenges in Integration of Heterogeneous Internet of Things,” *Sci. Program.*, vol. 2022, p. e8626882, Aug. 2022, doi: 10.1155/2022/8626882.

[30] S. Khan, S. Parkinson, and Y. Qin, “Fog computing security: a review of current applications and security solutions,” *J. Cloud Comput.*, vol. 6, no. 1, p. 19, Aug. 2017, doi: 10.1186/s13677-017-0090-3.

[31] A. A.-N. Patwary *et al.*, “Towards Secure Fog Computing: A Survey on Trust Management, Privacy, Authentication, Threats and Access Control,” *Electronics*, vol. 10, no. 10, Art. no. 10, Jan. 2021, doi: 10.3390/electronics10101171.

[32] R. Garg, Sz. Varadi, and A. Kertesz, “Legal Considerations of IoT Applications in Fog and Cloud Environments,” in *2019 27th Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP)*, Feb. 2019, pp. 193–198. doi: 10.1109/EMPDP.2019.8671620.

[33] Z. Ashi, M. Al-Fawa’reh, and M. Al-Fayoumi, “Fog Computing: Security Challenges and Countermeasures,” *Int. J. Comput. Appl.*, vol. 175, no. 15, pp. 30–36, Aug. 2020.

[34] R. Rezapour, P. Asghari, H. H. S. Javadi, and S. Ghanbari, “Security in fog computing: A systematic review on issues, challenges and solutions,” *Comput. Sci. Rev.*, vol. 41, p. 100421, Aug. 2021, doi: 10.1016/j.cosrev.2021.100421.

[35] M. A. Wright, “The Advanced Encryption Standard,” *Netw. Secur.*, vol. 2001, no. 10, pp. 11–13, Oct. 2001, doi: 10.1016/S1353-4858(01)01018-2.

[36] F. Elwy, R. Aburukba, and A. R. Al-Ali, “Role of Fog Computing in Smart Spaces,” in *2022 IEEE International Conference on Edge Computing and Communications (EDGE)*, Jul. 2022, pp. 69–76. doi: 10.1109/EDGE55608.2022.00021.

[37] L. M. Vaquero and L. Rodero-Merino, “Finding your Way in the Fog: Towards a Comprehensive Definition of Fog Computing: ACM SIGCOMM Computer Communication Review: Vol 44, No 5,” Oct. 2014. https://dl.acm.org/doi/10.1145/2677046.2677052 (accessed Jul. 01, 2023).

[38] I. Ud Din *et al.*, “The Internet of Things: A Review of Enabled Technologies and Future Challenges,” *IEEE Access*, vol. 7, pp. 7606–7640, 2019, doi: 10.1109/ACCESS.2018.2886601.

[39] S. Parikh, D. Dave, R. Patel, and N. Doshi, “Security and Privacy Issues in Cloud, Fog and Edge Computing,” *Procedia Comput. Sci.*, vol. 160, pp. 734–739, Jan. 2019, doi: 10.1016/j.procs.2019.11.018.

[40] A. M. Alwakeel, “An Overview of Fog Computing and Edge Computing Security and Privacy Issues,” *Sensors*, vol. 21, no. 24, p. 8226, Dec. 2021, doi: 10.3390/s21248226.

[41] A. Liebl, “Authentication in distributed systems: a bibliography,” *ACM SIGOPS Oper. Syst. Rev.*, vol. 27, no. 4, pp. 31–41, Oct. 1993, doi: 10.1145/163640.163643.

[42] A. Moore, “Authenticity as authentication,” *Pop. Music*, vol. 21, no. 2, pp. 209–223, May 2002, doi: 10.1017/S0261143002002131.

[43] Y. Atwady and M. Hammoudeh, “A Survey on Authentication Techniques for the Internet of Things,” in *Proceedings of the International Conference on Future Networks and Distributed Systems*, in ICFNDS ’17. New York, NY, USA: Association for Computing Machinery, Jul. 2017. doi: 10.1145/3102304.3102312.

[44] M. A. Siddiqi, H. Yu, and J. Joung, “5G Ultra-Reliable Low-Latency Communication Implementation Challenges and Operational Issues with IoT Devices,” *Electronics*, vol. 8, no. 9, Art. no. 9, Sep. 2019, doi: 10.3390/electronics8090981.

[45] C. Silva, J. P. Barraca, and R. Aguiar, “eSIM suitability for 5G and B5G enabled IoT verticals,” in *2021 8th International Conference on Future Internet of Things and Cloud (FiCloud)*, Aug. 2021, pp. 210–216. doi: 10.1109/FiCloud49777.2021.00038.

[46] Cisco, “5G Network Architecture - Cisco.” https://www.cisco.com/c/en/us/solutions/service-provider/5g-network-architecture.html (accessed Jul. 01, 2023).

[47] R. Dangi, P. Lalwani, G. Choudhary, I. You, and G. Pau, “Study and Investigation on 5G Technology: A Systematic Review,” *Sensors*, vol. 22, no. 1, p. 26, Dec. 2021, doi: 10.3390/s22010026.

[48] S. Papavassiliou, “Software Defined Networking (SDN) and Network Function Virtualization (NFV),” *Future Internet*, vol. 12, no. 1, Art. no. 1, Jan. 2020, doi: 10.3390/fi12010007.

[49] H. Yang, T. So, and Y. Xu, “Chapter 12 - 5G network slicing,” in *5G NR and Enhancements*, J. Shen, Z. Du, Z. Zhang, N. Yang, and H. Tang, Eds., Elsevier, 2022, pp. 621–639. doi: 10.1016/B978-0-323-91060-6.00012-X.

[50] A. Mishra, A. Swain, A. K. Ray, and R. M. Shubair, “Chapter 5 - Convergent network architecture of 5G and MEC,” in *5G IoT and Edge Computing for Smart Healthcare*, A. K. Bhoi, V. H. C. de Albuquerque, S. N. Sur, and P. Barsocchi, Eds., in Intelligent Data-Centric Systems. Academic Press, 2022, pp. 111–138. doi: 10.1016/B978-0-323-90548-0.00003-6.

[51] Microsoft, “What Is Edge Computing? | Microsoft Azure.” https://azure.microsoft.com/en-ca/resources/cloud-computing-dictionary/what-is-edge-computing/ (accessed Jul. 01, 2023).

[52] A. Hazarika and M. Rahmati, “Towards an Evolved Immersive Experience: Exploring 5G- and Beyond-Enabled Ultra-Low-Latency Communications for Augmented and Virtual Reality,” *Sensors*, vol. 23, no. 7, Art. no. 7, Jan. 2023, doi: 10.3390/s23073682.

[53] CableLabs, “A Comparative Introduction to 4G and 5G Authentication,” *CableLabs*, 2019. https://www.cablelabs.com/insights/a-comparative-introduction-to-4g-and-5g-authentication (accessed Apr. 25, 2023).

[54] Xelu86, “Extensible Authentication Protocol (EAP) for network access in Windows,” Jun. 19, 2023. https://learn.microsoft.com/en-us/windows-server/networking/technologies/extensible-authentication-protocol/network-access (accessed Jul. 01, 2023).

[55] C. Allen and T. Dierks, “The TLS Protocol Version 1.0,” Internet Engineering Task Force, Request for Comments RFC 2246, Jan. 1999. doi: 10.17487/RFC2246.

[56] P. Bellavista, J. Berrocal, A. Corradi, S. K. Das, L. Foschini, and A. Zanni, “A survey on fog computing for the Internet of Things,” *Pervasive Mob. Comput.*, vol. 52, pp. 71–99, Jan. 2019, doi: 10.1016/j.pmcj.2018.12.007.

[57] Y. Guo and Y. Guo, “FogHA: An efficient handover authentication for mobile devices in fog computing,” *Comput. Secur.*, vol. 108, p. 102358, Sep. 2021, doi: 10.1016/j.cose.2021.102358.

[58] J. Hu, Z. Li, P. Li, and J. Liu, “A Lightweight and Secure Authentication Protocol for 5G mMTC,” in *2022 IEEE 9th International Conference on Cyber Security and Cloud Computing (CSCloud)/2022 IEEE 8th International Conference on Edge Computing and Scalable Cloud (EdgeCom)*, Jun. 2022, pp. 195–200. doi: 10.1109/CSCloud-EdgeCom54986.2022.00041.

[59] F. Abdullah, D. Kimovski, R. Prodan, and K. Munir, “Handover authentication latency reduction using mobile edge computing and mobility patterns,” *Computing*, vol. 103, no. 11, pp. 2667–2686, Nov. 2021, doi: 10.1007/s00607-021-00969-z.

[60] A. Yazdinejad, R. M. Parizi, A. Dehghantanha, and K.-K. R. Choo, “Blockchain-Enabled Authentication Handover With Efficient Privacy Protection in SDN-Based 5G Networks,” *IEEE Trans. Netw. Sci. Eng.*, vol. 8, no. 2, pp. 1120–1132, Apr. 2021, doi: 10.1109/TNSE.2019.2937481.

[61] A. Yang, J. Weng, K. Yang, C. Huang, and X. Shen, “Delegating Authentication to Edge: A Decentralized Authentication Architecture for Vehicular Networks,” *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 2, pp. 1284–1298, Feb. 2022, doi: 10.1109/TITS.2020.3024000.

[62] H. Zhong, L. Chen, J. Cui, J. Zhang, I. Bolodurina, and L. Liu, “Secure and Lightweight Conditional Privacy-Preserving Authentication for Fog-Based Vehicular Ad Hoc Networks,” *IEEE Internet Things J.*, vol. 9, no. 11, pp. 8485–8497, Jun. 2022, doi: 10.1109/JIOT.2021.3116039.

[63] R. Kalaria, A. S. M. Kayes, W. Rahayu, and E. Pardede, “A Secure Mutual authentication approach to fog computing environment,” *Comput. Secur.*, vol. 111, p. 102483, Dec. 2021, doi: 10.1016/j.cose.2021.102483.

[64] E. Lara, L. Aguilar, M. A. Sanchez, and J. A. García, “Lightweight Authentication Protocol for M2M Communications of Resource-Constrained Devices in Industrial Internet of Things,” *Sensors*, vol. 20, no. 2, Art. no. 2, Jan. 2020, doi: 10.3390/s20020501.

[65] M. Han, S. Liu, S. Ma, and A. Wan, “Anonymous-authentication scheme based on fog computing for VANET,” *PLOS ONE*, vol. 15, no. 2, p. e0228319, Feb. 2020, doi: 10.1371/journal.pone.0228319.

[66] C.-M. Chen, B. Xiang, Y. Liu, and K.-H. Wang, “A Secure Authentication Protocol for Internet of Vehicles,” *IEEE Access*, vol. 7, pp. 12047–12057, 2019, doi: 10.1109/ACCESS.2019.2891105.

[67] Y. Yao, X. Chang, J. Mišić, V. B. Mišić, and L. Li, “BLA: Blockchain-Assisted Lightweight Anonymous Authentication for Distributed Vehicular Fog Services,” *IEEE Internet Things J.*, vol. 6, no. 2, pp. 3775–3784, Apr. 2019, doi: 10.1109/JIOT.2019.2892009.

[68] Maged Hamada Ibrahim, “Octopus: An Edge-Fog Mutual Authentication Scheme,” *Int. J. Netw. Secur.*, vol. 18, no. 6, Nov. 2016, doi: 10.6633/IJNS.201611.18(6).10.

[69] X. Jia, D. He, N. Kumar, and K.-K. R. Choo, “Authenticated key agreement scheme for fog-driven IoT healthcare system,” *Wirel. Netw.*, vol. 25, no. 8, pp. 4737–4750, Nov. 2019, doi: 10.1007/s11276-018-1759-3.

[70] X. Jia, D. He, N. Kumar, and K.-K. R. Choo, “A Provably Secure and Efficient Identity-Based Anonymous Authentication Scheme for Mobile Edge Computing,” *IEEE Syst. J.*, vol. 14, no. 1, pp. 560–571, Mar. 2020, doi: 10.1109/JSYST.2019.2896064.

[71] F. Dewanta and M. Mambo, “A Mutual Authentication Scheme for Secure Fog Computing Service Handover in Vehicular Network Environment,” *IEEE Access*, vol. 7, pp. 103095–103114, 2019, doi: 10.1109/ACCESS.2019.2931217.

[72] A. Koutsos, “The 5G-AKA Authentication Protocol Privacy,” in *2019 IEEE European Symposium on Security and Privacy (EuroS&P)*, Jun. 2019, pp. 464–479. doi: 10.1109/EuroSP.2019.00041.

[73] M. Abdrabou, D. El-Wanis, and A. Elbayoumy, “Security Enhancement for LTE Authentication Protocol (EPS-AKA),” *Int. Conf. Aerosp. Sci. Aviat. Technol.*, vol. 16, pp. 1–10, May 2015, doi: 10.21608/asat.2015.23028.

[74] S. Behrad, E. Bertin, and N. Crespi, “Securing authentication for mobile networks, a survey on 4G issues and 5G answers,” in *2018 21st Conference on Innovation in Clouds, Internet and Networks and Workshops (ICIN)*, Feb. 2018, pp. 1–8. doi: 10.1109/ICIN.2018.8401619.

[75] N. Saxena, S. Grijalva, and N. S. Chaudhari, “Authentication Protocol for an IoT-Enabled LTE Network,” *ACM Trans. Internet Technol.*, vol. 16, no. 4, p. 25:1-25:20, Dec. 2016, doi: 10.1145/2981547.

[76] R. Borgaonkar, L. Hirschi, S. Park, and A. Shaik, “New Privacy Threat on 3G, 4G, and Upcoming 5G AKA Protocols,” *Proc. Priv. Enhancing Technol.*, vol. 2019, no. 3, p. 108, Jul. 2019, doi: 10.2478/popets-2019-0039.

[77] S. Kwon, S. Park, H. Cho, Y. Park, D. Kim, and K. Yim, “Towards 5G-based IoT security analysis against Vo5G eavesdropping,” *Computing*, vol. 103, no. 3, pp. 425–447, Mar. 2021, doi: 10.1007/s00607-020-00855-0.

[78] T. Fei and W. Wang, “The vulnerability and enhancement of AKA protocol for mobile authentication in LTE/5G networks,” *Comput. Netw.*, vol. 228, p. 109685, Jun. 2023, doi: 10.1016/j.comnet.2023.109685.

[79] Symmetry Electronics Team, “From Telit: 5G IoT Security Issues: A Guide to Next-Gen Wireless Network Risks | Symmetry Electronics,” Jul. 24, 2019.

https://www.symmetryelectronics.com/blog/5g-iot-security-issues-a-guide-to-next-gen-wireless-network-risks/ (accessed Jun. 26, 2023).

[80] IBM, “IBM Documentation,” Mar. 29, 2023. https://www.ibm.com/docs/en/sdk-java-technology/8?topic=handshake-tls-13-protocol (accessed Apr. 25, 2023).

[81] D. Grant, “TLS 1.3 is going to save us all, and why IoT is still insecure,” *The Cloudflare Blog*, Dec. 24, 2017. http://blog.cloudflare.com/why-iot-is-insecure/ (accessed Jun. 26, 2023).

[82] G. Restuccia, H. Tschofenig, and E. Baccelli, “Low-Power IoT Communication Security: On the Performance of DTLS and TLS 1.3,” in *2020 9th IFIP International Conference on Performance Evaluation and Modeling in Wireless Networks (PEMWN)*, Dec. 2020, pp. 1–6. doi: 10.23919/PEMWN50727.2020.9293085.

[83] E. Ahmed *et al.*, “Bringing Computation Closer toward the User Network: Is Edge Computing the Solution?,” *IEEE Commun. Mag.*, vol. 55, no. 11, pp. 138–144, Nov. 2017, doi: 10.1109/MCOM.2017.1700120.

[84] L. M. Vaquero, L. Rodero-Merino, J. Caceres, and M. Lindner, “A break in the clouds: towards a cloud definition,” *ACM SIGCOMM Comput. Commun. Rev.*, vol. 39, no. 1, pp. 50–55, Dec. 2009, doi: 10.1145/1496091.1496100.

[85] P. M. Mell and T. Grance, “The NIST Definition of Cloud Computing,” *NIST*, Sep. 2011, Accessed: Aug. 27, 2023. [Online]. Available: https://www.nist.gov/publications/nist-definition-cloud-computing

[86] “Mobile edge computing, Fog et al.: A survey and analysis of security threats and challenges - ScienceDirect.” https://www-sciencedirect-com.proxy1.lib.uwo.ca/science/article/pii/S0167739X16305635 (accessed Aug. 28, 2023).

[87] L. F. Bittencourt, J. Diaz-Montes, R. Buyya, O. F. Rana, and M. Parashar, “Mobility-Aware Application Scheduling in Fog Computing,” *IEEE Cloud Comput.*, vol. 4, no. 2, pp. 26–35, Mar. 2017, doi: 10.1109/MCC.2017.27.

[88] Q. Qi and F. Tao, “A Smart Manufacturing Service System Based on Edge Computing, Fog Computing, and Cloud Computing,” *IEEE Access*, vol. 7, pp. 86769–86777, 2019, doi: 10.1109/ACCESS.2019.2923610.

[89] M. Qi, J. Chen, and Y. Chen, “A secure authentication with key agreement scheme using ECC for satellite communication systems,” *Int. J. Satell. Commun. Netw.*, vol. 37, no. 3, pp. 234–244, 2019, doi: 10.1002/sat.1279.

[90] D. Liu, P. Ning, and R. Li, “Establishing pairwise keys in distributed sensor networks,” *ACM Trans. Inf. Syst. Secur.*, vol. 8, no. 1, pp. 41–77, Feb. 2005, doi: 10.1145/1053283.1053287.

[91] J. Srinivas, A. K. Das, N. Kumar, and J. J. P. C. Rodrigues, “Cloud Centric Authentication for Wearable Healthcare Monitoring System,” *IEEE Trans. Dependable Secure Comput.*, vol. 17, no. 5, pp. 942–956, Sep. 2020, doi: 10.1109/TDSC.2018.2828306.

[92] J. Hermans, F. Vercauteren, and B. Preneel, “Speed Records for NTRU,” in *Topics in Cryptology - CT-RSA 2010*, J. Pieprzyk, Ed., in Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 2010, pp. 73–88. doi: 10.1007/978-3-642-11925-5_6.

[93] S. Zhang, X. Du, and X. Liu, “A novel and quantum-resistant handover authentication protocol in IoT environment,” *Wirel. Netw.*, vol. 29, no. 6, pp. 2873–2890, Aug. 2023, doi: 10.1007/s11276-023-03342-4.

Curriculum Vitae

Name:	Jiayi Zhang
Post-secondary Education and Degrees:	The University of Western Ontario London, Ontario, Canada 2017-2021 B. Computer Science The University of Western Ontario London, Ontario, Canada 2021-2023 M.E.Sc.
Honours and Awards:	University of Western Ontario Administration Scholarship 2017-2018
Related Work Experience	Teaching Assistant The University of Western Ontario 2021-2023