Faculty of Engineering and the Built Environment

Department of Electronic and Computer Engineering

Privacy for D2D Communications Based Applications and Services in IoT Enabled Networks

BOPAPE LP (21029737)

A thesis submitted in fulfillment of the requirements

for the Master of Engineering Degree

November 2021

Supervisor: Prof B Nleya

Signature: Date:___04 November, 2021_____

Acknowledgment

My sincere thanks to the Almighty, family and friends for the unconditional as well as unconditioned support throughout pursuing this work. Also, appreciate the unconditional guidance from the Academic Supervisor throughout this journey. I also extend my gratitude to peers in the Postgraduate Laboratory for constantly encouraging me throughout my research work. Last but not least l once again direct my special thanks to the Almighty and family.

Declaration

I hereby declare that this submission is my own work and to the best of my knowledge it neither contains materials previously published or written by another person, nor material which to a major extent has been accepted for the award of asimilar qualification/ degree at the Durban University of Technology or any other educational institution. I also declare that the intellectual content of this thesis is a product of my work. Any contribution made to the research by others especially in the use of equipment for sample analysis has been explicitly acknowledged in the dissertation. Further, I have acknowledged all sources used and have cited these in the reference section.

Bopape L.P

____03/11/2021_____

Date

Abstract

With the advent of IoT, Device-to-Device (D2D) communications has afforded a new paradigm that reliably facilitates data exchange among devices in proximity without necessarily involving the base (core) network. It is geared towards the need to improve network performance where short-range communications is concerned, as well as supporting proximity-based services. However, the relentless growth in the number of network end-users as well as interconnected communication-capable devices, in the next-generation IoT-based 5G cellular networks has resulted in novel services and applications, most of which are security-sensitive. It is thus of paramount importance that security issues be addressed. A posing challenge is that the devices are mostly resource-constrained in both power and computing. As such, it is not practical to implement present day as well as traditional security frameworks and protocols under such a scenario, unless strides are taken towards the improvements of data throughput rates, higher bandwidth provisioning, lower round trip latencies, enhanced spectral efficiencies, and energy efficiency (leading to even lower power consumption, by the already constrained devices) in IoT 5G/LTE networks. Therefore, this work focused on exploring and designing schemes that enhance security and privacy among communicating parties. Otherwise, without reliable as well as robust privacy and security preservation measures in the network, most services and applications will be exposed to various forms of malicious attacks. With such a widened cyber-attack space, both privacy and security for end users can easily be compromised.

The work herein addresses privacy for subscribers to the various available services and applications as well as security of the associated data. Ultimately, we propose a Fog-Cloud computing paradigm-assisted security framework that comprises two schemes. The aim is to implement a lightweight-based cartographic algorithm that ensures that communication overheads, round trip latencies, computational loads as well as energy consumption by the otherwise resource-constrained surveillance cameras deployed remotely, are kept minimal. Overall, by way of both analysis and simulation, we ascertain that a Fog-Cloud computing-based lightweight security-based scheme has the potential to greatly improve security and privacy preservation, as well as overall performance despite the resource-constrained nature of the devices.

Table of Contents

List of Figures

List of Tables

List of abbreviations

D2D – Device to Device

IoT – Internet of Things

ITU- International Telecommunications Union

IP – Internet Protocol

DoS - Denial of Service

UE – User Devices

LTE – Long Term Evolution

LTE-A – Long Term Evolution Advanced

ISM – Industrial Scientific and Medical

MANET - Mobile Ad-hoc Networks

FAC – Fine-grained Access Control

PC – Personal Computer

AKA – Authentication and Key Agreement

RFID – Radio Frequency Identification

EPC – Data Encryption Standard

DHKE – Diffie Hellman Key exchange

SMD – Secure Message Delivery

GKA – Group Key Agreement

MITMA – Man in the Middle Attack

DDoS – Distributed Denial of Service

Au - Authentication

Pr – Private

MCC - Mobile cloud computing

QoT - Quality of Transmission

ECDH - Elliptic Curves Diffie-Hellman

PSN - Public Switched Network

UEs - Equipments

AP - Access Point

EPC - Evolved Packet Core

AVISPA - Automated Validation of Internet Security Protocols and Applications

HLPSL - High-Level Protocol Language

OFMC - On-the-fly-Model-Checker

CL-AtSe - Constraint-Logic- based Attack Searcher

BAN - Body Area Network

HUP - Hospital Uploading

PUP - Patient Uploading

TP - Treatment and Prescription

CP - Routing Checkup

HMAC - Hash Message Authentication Code

IBS - Identity-Based Signatures

IBE - Identity-Based Encryption

ECDLP - Elliptic Curve Discrete Logarithm Problem

CLGSC - Certificate Less Encryption Scheme

HSS - Home Subscriber Server

EPC - Evolved Packet Core

eNB - Evolved Node B

IMSI - International Mobile Subscriber Identity

1. Introduction

1.1 Overview Summary

The evolution of all legacy networks into the resurging Internet-of-Things (IoT) that ultimately provides interconnectivity among various objects and devices worldwide is gradually becoming practical. There is a potential of interconnecting more than 23 billion objects and devices by the year 2025. [1]. Most of these are derived from existing corporate networks currently provisioning connectivity to billions of IP (version 4 or 6) compliant devices, emerging IoT-based Industry automation, GSM (smart homes), security-related surveillance systems such as IP cameras, e.tc. All the aforementioned devices and systems are geared towards bettering the everyday lives of existing mankind as well as directly improving the overall lifestyles of both users and the rest of the population. Key to the successful operation of future generation networks such as the envisaged fully fletched IoTs would be enabling efficient secured communication protocols and architectures that interconnect the various distributed and mostly sparsely located systems cons tied by the devices and objects.

The International Telecommunications Union (ITU) is the main regulator, defines an IoT-enabled network as being a worldwide deployed information dedicated that facilitates multitudes and services and applications advanced services through interconnected devices (both physical and virtual) based on current and ever-evolving interoperable protocols and technologies. This massive advantage of IoT devices and objects interconnectivity bears a cost of heightened security and privacy for both users and components themselves. It is noted that the majority of the IoT devices and objects are designed and manufactured independently by large corporations, teams without relying fully upon cooperating in taking into account the security and privacy issues.

Often the various parties are constrained with regards to enabling resources as well as delivery timelines for developing the components and launching them in the ever competing markets. Tight schedules (timelines) and limited enabling resources often force the manufacturers to eventually opt for corner-cutting (i.e. commissioning and using improperly tested code snippets and as well as not following procedural design principles with regards to security [1]. Most users and proprietary vendors may not pay close attention to security and privacy risks such as vulnerabilities as well as susceptibilities to attacks (using IoT compliant devices and related components). Often the proprietary vendors take a proactive approach in addressing

and security deficiencies in their products, i.e., by providing updates and security. However, these approaches often provide temporary remedies for most of the security and privacy threats.

One of the reasons is that of the complex nature of Device-to-Device (D2D) interactions in the IoT arena [2]. Often, an IoT-based service or application may involve a set of installed and networked devices and objects specialized for collaboratively performing a particular task. This could include tasks such as weather forecasting or CCTV surveillance in a particular vicinity. The collaborative interaction as well as is interdependency among the group of devices can be capitalized upon by attackers to gain access to the entire system. An example would be an intruder (attacker) triggering a false signal to a CCTV surveillance system to open a carport door thus giving him access to a user's main house, as the two are often attached side-by-side, and with a linking door to the kitchen or main lounge. Most IoT devices and objects (e.g., CCTV cams, smart locks, motion sensors) are often installed outside the house, thus readily making them vulnerable to illegal tampering. It is thus imperative to limit as well as bound these D2D linkages to reduce the risks for attackers to trickily destabilize trick the system into helping adversaries.

Traditionally, firewalling has always aided in strengthening security and privacy since the aggregate incident traffic can be monitored and any anomalies or peculiar behaviors be blocked. Note however that D2D communications interaction may be problematic to monitor as it often uses an "out of band" communication, hence the need for an approach that can contextualize as well as take into account D2D dependencies for securing network communications. Moreover, the fire-walling approach being reactive in nature implies that a threat has to be detected first before enforcing any security measures. The heterogeneous nature of current IoT compliant systems is because often each system consists of multitudes of devices from various vendors or manufacturers with diversities in firmware. The diverse nature of firmware models, shortage of energy and hardware resources, software application stack, lack of security patches and software updates, etc, make it problematic to develop address security issues at the network periphery using software security applications for the IoT devices, objects, and systems.

The seeming lack of growth truth is also a huge problem. e.g., most anti-virus software utilize massive databases of attack signatures and malware to identify potential anomalous behavior from network traffic traces as well as system activities. As previously cited the diversity of

firmware, application stacks, and protocols, etc. makes it problematic to develop signature databases for IoT users. Key D2D dependencies-related information is required for developing a scalable security solution for IoT systems. For some particular systems, devices may leave, or new ones may be added. For that reason, it becomes difficult to develop a permanent configuration model in terms of policies given for D2D communication as it is complex plus there are many devices involved. Moreover, such setups require frequent updating as devices join/vacate the network.

Overall, the various IoT systems and services can only be e secure provided the associated D2D communication sessions are secured themselves. Because the majority of would-be D2D communication-based services and applications will be based on group collaborative interactions, it may be necessary to advocate for group-based authentication and key agreement (AKA) protocol approaches for achieving effective authentication. These will be expected to satisfy basic security requirements. Examples of such security requirements include mutual authentication, confidentiality. Summarily the key security-related challenges in IoT elated communications include but are not limited to:

- User security. e.g. user devices such as smart meters autonomously frequently relay data to the Utility center and as such, there is a risk of the exchanged data being intercepted by intruders to figure out devices being used in the home, infer consumer's activities, as well as times when the home is vacated.
- The multitudes of intelligent IoT network pillar devices may eventually be used as attack entry points. Furthermore, the vast areas of coverage by the smart grid itself make control and operation quite complex.
- Physical security: Most components that build the various IoT systems, applications, and services are placed in insecure places such as outside buildings. As such the fact that there are many insecure physical locations makes SGs vulnerable to physical attacks.
- The lifetime span disparities between power systems components versus those of IoT-driven technologies and infrastructures implies that the two exist alongside the older equipment acting as a weak entry point for malicious attacks. The old equipment may also be incompatible with the current power system devices hence compromising overall delivery efficiencies.

- Implicit trust relationships among traditional power devices make D2D communication susceptible to malicious attacks such as data spoofing. This is because a single device's condition may adversely affect the operations of its peers.
- Disparities in Team's background may lead to considerable vulnerabilities. This is attributed to frequent uncoordinated communication between teams which leads to lots of bad decisions as well.
- The utilizing of IP-compatible equipment in the IoT infrastructure makes it susceptible to traditional network attacks.
- The conglomerating of several stakeholders (investors) in the IoT network, might give rise to insider attacks."

1.2 Background to D2D Communications

D2D communication is primarily concerned with direct communications between devices in proximity and without the direct support of any formalized network infrastructures. In that way, D2D communications can facilitate direct linkages as well as data exchanges among two or more D2D communication compliant devices. Typical usage examples would be in tablets, network printers, IP cameras, or connected vehicles using common technologies such as WiFi Direct and Bluetooth.

However, the surge in the numbers of connected networked devices and objects has also led to the introduction of next-generation wireless communication technologies such as 5G/7G that are expected to provide the required interconnectivity support. In that way, more innovative applications and services will continue to be introduced to IoT with a highly improved quality of service (QoS) to users. Typical QoS primitives will include higher data throughput rates, overall higher channel bandwidths, reduced roundtrip network latencies, lower power consumption, and overall enhanced spectral efficiencies. Numerous standardization bodies have advocated for D2D communication as the underlying key element in 5G technology as this will ensure efficient spectrum reuse and consequently improved spectral efficiency. The same communication technology takes advantage of proximities among devices and objects to further improve renderable QoS. However, quite a few design and technical issues have arisen that require adequate addressing. These include new device discovery modes and procedures, wireless network resource management schemes, and physical layer architecture. Efforts towards standardization in D2D communications have gained momentum since 2014. One such effort is by the 3rd Generation Partnership Project (3GPP) which has since defined a set of

directives and enhancements to the 5G/LTE architectural framework in a bid to facilitate the deployment of Proximity-based Services (ProSe).

1.3 D2D Related Technologies

D2D communications have been existing for decades in mobile services which are based on technologies such as, Bluetooth and Wi-Fi Direct both of which only cover short ranges. Even though these technologies bring up some limits, it appears to be no longer appropriate for the generations to come of the proximity-aware, current time, and context-aware services intended in the context of evolution to 5G network. Direct communication technology may not be a new concept. This is because already there exist several other technologies that facilitate UEs in proximity to exchange data.[2], [3]. However, there are various emerging challenges with regards to its direct implementation in next and future-generation wireless networks such as 5G/7G. These are summarized as follows:

- *D2D's compatibilities with NFV and SDN*: D2D communication must be completely compatible with existing and emerging core network infrastructure and applications such as virtual network function (NFV) and software-defined network (SDN). It is, therefore, necessary that new D2D communication protocols be implemented to harmonize operations with future core networks [4].
- *D2D with millimeter wave (mmWave)*, mmWave is proposed for future generation wireless networks for improved QoS support as it facilitates UEs operating at high frequencies. Whereas it targets short-range communications and is hence suitable for D2D communications, it however has an intricate propagation model which may not be easy to blend with D2D.
- *D2D's coexistence with future hyper-dense small cells*. At the present moment, research assumes D2D coexists with macro cells. However, it may be necessary to consider hyper-dense small cells as mutual interference will become more pronounced.

1.4 Security and Privacy Challenges in D2D communications

Addressing both privacy and security is quite a key issue in regards to the successful deployment of D2D communication-based services in the network. The resource-constrained nature of the devices in the IoT networks means that the traditional security and privacy pro-

tocols can not be applied directly since they are intensive resource demanding. Typically, they will require more memory and processing power. The IoT network's deployed communicating devices and objects are often operated in adverse environments, thus being readily vulnerable and susceptible to both physical and semantic security attacks. An important aspect of the successful running of D2D communication-based services and applications is that they are reliably authenticated. A typical example service or application will involve several devices collaborating as a group. In such a scenario, mutual authentication within the group is mandatory to exclude intruding actions by adversaries.

Group-based AKA protocols are thus necessitated to efficiently carry out the necessary mutual authentication. Efficient in the sense that the computational, as well as communication overhead loads, must be kept at a minimum because of the resource-constrained nature of the devices themselves. It is always mandatory that key security requirements be maintained. Normally a mutually agreed single security (encryption) key would be utilized for communication purposes. Such an algorithm should achieve efficacy during operations despite the adverse conditions in which the devices are operating as earlier alluded to.

The overall aim is to ensure identity privacy, as well as security for the data, are not compromised. The multi-domain nature of a typical IoT network means that a given service or application may run in the form of several deployed groups, spanning over a few network domains. This will require linking them across the domains which are independently administered. They are independently administered in the sense that the security policies might differ from one network domain to another. The vast geographical spread may contribute adversely to the maximum acceptable delay latencies.

With the aforesaid issues and challenges, we will propose and investigate a group authentication as well as a key agreement protocol for D2D communications that is assumed to operate among multiple domains as well as operators. In this case, we will advocate for cloud computing-based paradigms. However, ultimately, we will rely more on the Fog layer-assisted computing paradigm to reduce the end-to-end latencies as well as turnaround times. In so doing our focus is on ensuring the following:

- That the proposed security and privacy framework can carry out group authentication and as such all members of a group of participating devices are authenticated concurrently.

- That it preserves security as well as privacy requirements. In particular, the forward and backward secrecy must be preserved when a device vacates or joins a group.
- Taking cognizance of the resource-contained nature of the operational environment, we propose some lightweight forms of cryptography. We will also lean towards the usage of symmetric keys to keep computational as well as signaling overheads loads minimal.

1.5 Summary Research Problem

Based on the literature survey carried out so far, in this sub section, we provide a summary of the research problem already partly alluded to in the preceding section. Given the flexibility in connectivity provided by 5G wireless networks, and specifically the ability to achieve connectivity between pair devices without the aid of the base (core) network, the security risks likewise significantly widen. It follows that the cyber attack space correspondingly increases. In addressing possible solutions to countering various attacks, we need to take into cognizance the resource-constrained nature of most of the devices in the IoT space. As such in designing security schemes or protocols we should ensure the following:

- The scheme provides and satisfies both privacy and security requirements as already discussed in previous sections.
- That the designed scheme must incorporate algorithms that ensure minimized computations. In this regard, implementing some form of lightweight encryption approaches would reduce the computational burdens as fewer mathematical operations are involved.
- The schemes must minimize both signaling and communication overheads. The reason here is that by nature D2D communications by nature operate under very constrained link availabilities and as such both signaling and communication overheads must be kept at a minimum otherwise, the available link resources can quickly become congested.
- The scheme's turnaround times must be minimal. In short, end-to-end latencies within the network must be low. For that reason, Fog-Computing paradigm approaches could assist in decentralizing computations. E.g most of the computations can be done in the Fog layer, and only the complex ones escalated to the Cloud. The Fog-Computing aspect will also assist in providing adequate coverage at all times.

1.6 Main objectives

Based on the literature survey carried out so far, it is generally acknowledged that significant work on privacy and security for the various services and applications in the IoT arena has been accomplished. The incorporation of D2D communications in the 5G wireless networks enhances fail-safe network operation(s). However, it also intricates the design of privacy and security schemes as summarised in the previous section. We thus summarise the objectives of our core work as follows: -

- To carry out more literature survey on security, and privacy preservation for applications and services that may opt for D2D communications when the need arises.
- Investigate the efficacy of related protocols in this regard. The efficacy will mainly cover computational, signaling, communications overhead, latency, and overall energy efficiency in protocol operations.
- To study the general D2D communications protocols and IoT network architecture and related technologies.
- To acquaint adequately with both Cloud and Fog Layers-based computing paradigms. Ultimately the combined Fog-Cloud computing paradigm will be relied upon in proposing and evaluating a security framework.
- To propose and evaluate the efficacy of the Fog-Cloud paradigm-based security and privacy framework.

1.7 Outline

Chapter 2 explores the general IoTs architecture, and operational protocols, and standards.

Chapter 3 overviews D2D communications and standards.

Chapter 4: The chapter will focus on group security and privacy. A comprehensive survey and compilation of group-based authentication, as well as key agreement approaches, will also be incorporated in this same chapter.

Chapter 5 this chapter will focus on proposing a framework for enhancing privacy as well as security for D2D communications-based services and applications. A group authentication protocol will be investigated and evaluated.

Chapter 6 is a concluding chapter as well as spells out possible future research direction (s).

1.8 Contributions

The key contributions of the thesis are summarized as follows:

A comprehensive overview of D2D communication and related privacy issues. The work also discusses requirements for the fulfillment of privacy for D2D communications-based services and applications. An analysis of selected privacy authentication protocols for D2D communication is carried out. The work also proposes a Fog Computing paradigm-based Privacy/security framework for D2D based applications and services as it is seen to have the potential to drastically reduce latencies experienced with Cloud computing. The framework exploits the fog layer, which is the interfacing layer between the core and peripheral network sections to drastically reduce latencies as well as boost the limited computing powers in resource-constrained devices. It can also provide network context information which ultimately is used by fog applications and services to optimize context-awareness. Its support for location-awareness; means it can fully support device mobility which is a direct booster for location-based services and applications. Fog computing easily provides a local overview whereas a global overview will still be provided by cloud computing. Proposal of a security framework as well as a group authentication scheme (protocol) for D2D communication, based surveillance service.

1.9 Contributions

The contribution of this work is as follows:

[1]. A fully compiled 138 pp dissertation.

1.9.1 Journal Papers

[1]. L. Bopape, B Nleya, P. Khumalo and A. Mutsvangwa, “A Group Authentication And Data Security Scheme For Smart Metering In Smart Grids”, Ponte International Journal of Research and Sciences, Vol. 75, | No. 11/1 | January 2020 DOI: 10.21506/j.ponte.2020.1.4, ISSN: 0032-423 [72].

1.9.2 Refereed Conference Papers

[1]. P. Bopape, B. Nleya, and P. Khumalo, "A Privacy and Security Preservation Framework for D2D Communication Based Smart Grid Services," 2020 Conference on In-

formation Communications Technology and Society (ICTAS), Durban, South Africa, 2020, pp. 1-6, doi: 10.1109/ICTAS47918.2020.233995 [73].

[2] M. Molefe, B. Nleya, R. Chidzonga L. Bopape, and K. Sibiya, "An Energy-Efficient Impairment-Aware Routing Algorithm For Optical Transport Networks," 2021 Conference on Information Communications Technology and Society (ICTAS), Durban, South Africa, 2021, pp. 11-15, doi: 10.1109/ICTAS50802.2021.9395021 [78].

1.10 Summary Chapter Conclusions

- A summarized overview of D2D communication security challenges is highlighted.
- The main aims and objectives of the research are outlined.
- The envisaged general format of the dissertation is also outlined.

2. Internet of Things Architecture and Security Overview

2.1 Introduction

Legacy as well present-day networks are fast evolving to facilitate as well as accommodate new as well as other innovative applications and services. Typical examples will include the legacy IP and Power Smart Grid communications networks. This has resulted in diverse connectivity requirements as well as methodologies. A common networking platform called "The Internet of Things" (IoT) that has transformed the traditional IP and other data networks to D2D communication has taken renaissance. It aims to interconnect the entire universe seamlessly via a system of aiding devices, elements, objects, and sensors such that the result will be over 30 billion interconnected will be vigorously involved in data exchanges by 2022. This data represents services and applications that are key to mankind's general way of living. These will range from disaster early warning applications to telehealth services. The chapter looks at the current and envisaged IoT-enabled network architecture. Explicitly, we will describe a generalized IoT-enabled network's security architecture as the focus of the work herein on security and privacy. We then describe key aiding elements, standards, protocols, and spearheading alliances. Notably, we take particular attention to the presence and dominance of resource-constrained devices.

The paradigm shift from IP-based networks to IoT is expected to accelerate towards a peak by 2022 [6]. Advancements in enabling technologies together with mankind's desire to integrate the capabilities of ICT towards bettering his/her everyday life are perceived as the key accelerators to this paradigm shift. Essentially this Globe interconnectivity venture will benefit us all because of the services and applications that will use it as a platform. Besides health and disaster management, other areas that are likely to benefit from this venture extend to supply chain management, smart homes and cities, and education. At enabling technologies implementing level, it is noted that the diverse heterogeneity in resource availability, communication requirements, and hardware capabilities among the various objects will severely complicate the design and operational requirements in terms of resource provisioning. Notably at a hardware level, it would be necessary to address the variances in computing requirements, typical examples of which include memory, power, computation, or communication capabilities.

The variances in quality of transmission (QoT) and consequently Quality of Service (QoS) requirements, e.g., In quantifiable metrics such as latencies, data loss tolerances, and others is also a challenge towards realizing a common /universal network. E.g. devices require and expend lots of power when in computing mode, so constrained devices are likely to run out of the only available battery power whilst in computing mode, whereas that issue is never critical in devices that have reliable and abundant power supplies. The latter can always replenish or harvest extra power from the powering source(s) [7]. The contrasting characteristics make it problematic to design a network that satisfies both categories of devices. Adaptive cross-layer communication approaches are being explored to address this problem.

Partly the adaptive cross-layer communication approach is being implemented in various wireless networks such as sensor (WSNs), mesh (WMNs), and Ad-Hoc (AHNs) [8], [9]. The same is not practical for the IoT network because: -

- The variance in network topologies. IoT has centralized and hierarchical architectures, whereas AHNs, WSNs, and WMN networks mostly depict a flat topology in which nodes link in a hopping manner not necessarily involving the core IP network.
- Heterogeneity does not characterize in WSNs, since the nodes focus on a like goal, thus utilize similar communication protocols whereas IoT-enabled networks, there is significant heterogeneity in terms of resource requirements since goals are individual.

Besides security and privacy architectures for the IoT must be addressed. This is on the background of the heterogeneity and dynamicity of the IoT environment.

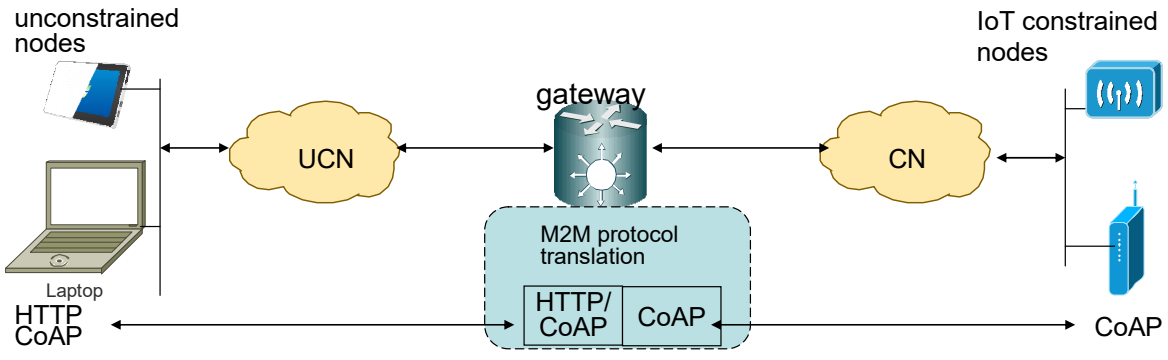


Figure 2.1: IoT heterogeneous space

Traditional l security controls are normally enforceable within a domain, however, the IoT will normally run an application or service whose associated devices are deployed in several domains. Besides the resource constrain factor characterizing most of these objects means

that most of the security protocols cannot be applied directly. Rather access control enforcement will also be problematic as associate control protocols are generally platform-specific and would not be cost-effective or generally feasible to implement in a multi-domain scenario. Figure 2.1 illustrates an IoT heterogenous space in which the machine-to-machine (M2M) protocols may require translation. Overall with the many criteria that an effective privacy and security framework must satisfy, it becomes rather worthwhile to focus more on D2D networking protocols rather than applying the existing ones directly.

- *Front-end Sensors and Equipment:* This facilitates the acquisition of data from sensors before r forwarding to designated central processing centers using D2D modules. The current setup has security compromised as they lack monitoring.
- *Network Denial of Service (DoS)*: The ability to deliver acceptable QoS may be problematic to realize as IoT-enabled networks directly facilitate overall D2D communication coordination/management which itself is susceptible to DoS attacks.
- *Back-end*: Back-end is incorporated in an IoT-enabled network system. It provides the required security and object (sensor) management functions.

Presently, secured links within an unconstrained node (UCN) domain can be facilitated using IPsec [10]. They however will not be able to link directly with constrained nodes (CN). This is attributed to the mismatching speeds in processing resources capabilities. Researchers have proposed partly delegating some of the loads to neighbor UCNs. In so doing, intermediary IoT Gateways can assist harmonizing and adapting the communications between the CN and UCN.

Note that the data is normally exchanged via public channels i.e via fixed PSTN or GSM links. Privacy however will require private channels so that the identity of the device or user is concealed throughout. In short: -

- *Privacy in Device*: Vulnerabilities to the unsecured device mean data can be siphoned and diverted. Therefore, is necessary to ensure that privacy is robust and reliable.
- *Privacy during Communication*: Privacy during device-to-device data exchanges can be maintained by way of using a set of secured communication protocols.
- *Privacy in Storage*: Enforcing anonymization and pseudonymization when accessing data storage facilities would help.

- *Privacy at Processing*: Digital Right Management (DRM) techniques can be used to ensure data privacy during processing. In this case, illegal use would be alleviated.

2.2 Devices, Elements, and Architecture Overview

Figure 2.2 shows the key IoT devices and components as specified by the ITU-T [11].

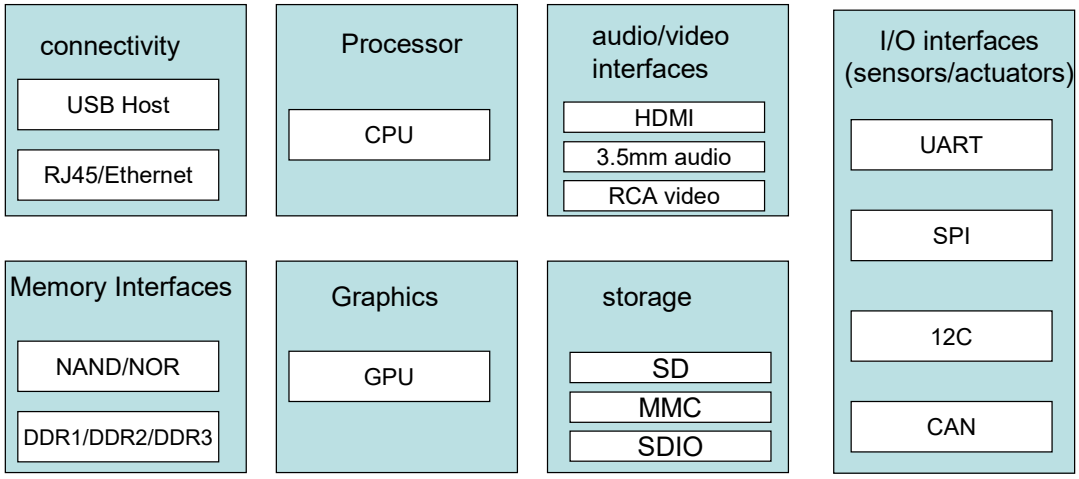


Figure 2.2: IoT devices and components

The architectural elements help in the realization of several functionalities that include:

- *Remote sensing.* This refers to the acquiring and processing of data after which key information is extracted and used as a reference in the future. The reference may be in the form of actions of remedial actions to be taken following an event occurrence.
- *Self-adapting utilities*: In this case, the deployed elements are provisioned with functionalities dynamically adapting to varying contexts. In this case, the associated objects will eventually align their actions accordingly.
- *Self-configuring*: This would apply to objects that acquire data from a changing environment. A typical example would be climatic sensors that may be expected to self-reconfigure when weather conditions change drastically.
- *Interoperable protocols*: The devices ought to be able to adapt and facilitate interoperability when the situation arises.
- *Self-Integrating*: This is to facilitate self-discovery or by peers. The goal is to have devices interacting harmoniously. Once discovered by peers, a device can now provide a self-description to their new peer devices or user applications.
- *Unique identity*: Every physical or virtual device is assigned a unique identity as well as an identifier. These elements may also be coupled/provisioned with context-

adaptable interfaces that also facilitate remote querying, monitoring, and control of associated devices.

Example functional architectural elements include communication, between devices application services, security, and privacy e.tc.

- *Communication*: This enables linkage and data exchanges between peers.
- *Services*: facilitating device control and modeling data analysis, and publishing as well as device discovery.
- *Management*: Utilizing various functions to ensure harmonious operation of the IoT.
- *Security*: Provisioning security-related services and functions.
- *Application:* Provisioning key modules for controlling and monitoring various aspects of the IoT system(s).

A consensus is slowly being reached towards an IoT-enabled works standardized architecture. The consensus among the various alliances and study groups seem to advocate for the following layers:-

- *Physical* (*perception) layer:* The layer comprising the various objects and devices. Mostly the deices are involved in data acquisitions.
- *Network layer:* This is the key layer to facilitating communication between peers within the IoT.
- *Transport layer:* The layer ensures safe data delivery between sender and destination.
- *Application layer:* This is the end-user interface for accessing various applications and services.
- *Processing layer*. Process data derived from the transport layer.
- *Enterprise /Business layer:* For the regulation of the entire IoT operations and this may include business, profit, and security models.

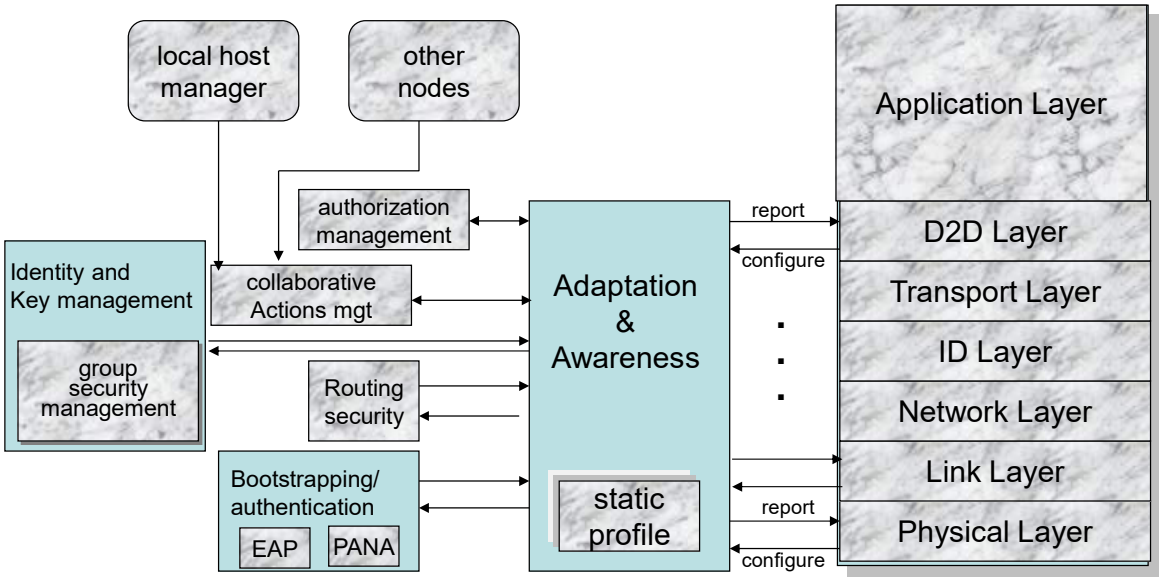


Figure 2.3: Generalized Secured Communications Architecture

Shown in Figure 2.3 is a generalized secured communications architecture model [10], [11]. On the right is the communications architectural layers, whilst the security layers are on the left side. As can be noted a D2D communication service layer is added after the user (applications) layer. Its role is to enable linkages between dissimilar network devices thus solving interoperability issues experienced with current M2M and equivalent technologies [12],[13].

- **Transport Layer's** functions are QoS, dependent, but retain the fundamental role of ensuring the process to process delivery of data.
- **The identification (ID)** layer's main function is to identify the required resources. It can facilitate privacy and authentication by way of utilizing the node ID. The Host Identity Protocol (HIP) can be relied upon by this layer
- **The network (NET)** layer is the equivalent of the IPng layer and takes care of logical addressing. And general end-to-end packet routing.
- **The MAC** layer regulates the utilization of channel resources. It will allocate access such that channel contentions are minimized [14].
- **Physical Layer (PHY) regulates** electrical specifications of the data-associated symbols. This includes appropriate line encoding for the various media

Security-wise, the following blocks are involved:-:

- **Bootstrapping and Authentication** control the incorporation of new nodes on the network. to the network. The Extensible Authentication Protocol (EAP) and the Pro-

tocol for carrying Authentication for Network Access (PANA) are key legacy protocols utilized on this layer [15], [116. The PANA also assists in improving overall interoperability.

- **Static Profile** negotiates with the endpoint(user) on what cryptographic protocol to use. This will be based on the end point's declared specifications such as its power source size, storage capacity, processing power, and desired security profile/preference.
- **Collaborative Actions Management** supports CNs that midway cannot cope up with computations as their resources may no longer sustain that. In this case, they seek, assistance from a neighboring CN.
- **Identity and Key Management** block regulates device privacy by way of using pseudonyms instead of real IDs.
- **The adaptation and Awareness** block helps identify the next node and which protocols would be compatible in interacting with it.
- **Group Security Management** provides multicast-related privacy at the Network layer level.
- **Routing Security block** acts as a resilient guard against malicious attacks. It is partly backed by the Local Trust Manager and Bootstrapping and Authentication modules.
- **Authorization Management (AuthZ Mgt.) It will normally** regulate authorization and access to available shared resources and other related services. It uses trust certificates for allowing entrance to resources. It can also decide whether entry can be granted without trust certificates.

2.3 Basic Functional Components

In a typical social IoT (SIoT) compliant network the devices and services can set up interrelationships among themselves as well as reconfigure from time to time. In that way, seamless among each other to achieve complex tasks in the form of services and applications can be achieved.

For such a model to work, it is a requirement to have various interoperating as well as interoperable components. Examples may include:-

Identification (ID): This is for object identification. This enables every object or device to be assigned a unique ID such as e.g.; a MAC address, IP address, or Universal Product Code. The overall purpose of an ID would be to facilitate the following:

Security controls: We will be able to place access restrictions to a given device, i.e. the owner (proprietor) will be able to restrict the type and quantities of devices that can connect to his/her device.

Service discovery: It becomes relatively easier for other devices to know what kind of services can be rendered by a particular identified device.

Relationship management: This facilitates devices knowing apriori which other devices they can connect to e.g. a cam coder screen connecting to a street surveillance camera.

Service composition: Typically such a module will facilitate will provide improved integrated services to users. An example is when a built-in geyser "energy consumption sensor" established a collaborative relationship with some analytics engine within the vicinity. In that way, it would be able to acquire lots of data regarding the energy usage patterns of the geyser. In that way, a comparison can be made with other geysers of the same capacity elsewhere, and eventually, a user may end up acquiring a more energy-efficient model geyser.

2.4 Representative Architecture.

Server-side architecture has been proposed for SIoTs. The servers typically interconnect all the components as well as aggregate all the renderable services and applications. From where users can access them. Its main layers from bottom to top, include the base, component as well as application layers.

The base layer stores a database of details and attributes of all the devices that connect to the server. Meta-information, and relationships associated data for the various devices may also be stored in the database. The component layer provides the necessary codes that are necessary to run the services and applications. Typically necessary universal driver software is also stored herein in case of connectivity incompatibilities among certain devices.

The application layer is a direct interface to the users from which the various services and applications can be accessed. At the device level, we have a couple of layers as well. The object layer, which facilitates device-to-device connectivity. In that way, devices would be able to connect via standardized protocols and hence exchange data/ information. It also acts as a direct interface to the social layer. The social layer regulates and manages the execution of various users' applications, executes queries, and also interfaces the application layer on the server.

2.5 Actuators and Sensors

Sensors are integral key components of smart devices and they act as data collectors for various applications and services. IoTs are characterized by context awareness, which will not be practically possible without sensors. Typically, IoT sensors miniaturized low cost, as well as low power devices. They often are power constrained as their battery sizes are often correspondingly small as well. In this subsection, we provide typical examples.

2.5.1 Mobile Phone-Based Sensors

Today's smartphones are equipped with several different types of sensors to facilitate various applications and services.

- The accelerometer sensor determines the motion as well as the acceleration of a GSM phone. It typically measures changes in speed in 3D. There are various types of accelerometers. E.g the gyroscope sensor is used to detect the GSM handset's orientation.

- The camera and microphone are typical sensors that capture audio and visual information respectively. The sensor will detect the earth's magnetic field thereby facilitating digital compassing as well as applications to detect the presence of metals and pinpoint precise locations.

- The GPS (Global Positioning System) sensors make use of GPS signals from orbiting satellites to detect a phone's location here on earth. Trilateration or triangulation is used to detect the location of the device. Typically, the GPS coordinates are e generated by three geo-orbiting satellites before being beamed to the earth.

- The light sensor manages the ambiance of light intensity on a GSM phone's screen and thus can be used to auto-regulate the screen's intensity as well as a contrast for optimum viewing by the user.
- The proximity sensor makes use of infrared (IR) LED, which emits IR rays. Which emit and cast rays to an object. The reflected rays can be used to compute the actual object's distance from the GSM phone. These same sensors can be used as event triggers when certain proximity to a phone is reached.
- Thermometers, barometers, and humidity sensors are also incorporated in some GSM brand phones to measure atmospheric pressure, humidity, and temperature.

- *Medical Sensors.* The IoT is quite beneficial for health care applications. The sensors acquire the critical data directly from our bodies before they relay it to medical specialists who will, in turn, give us feedback. A typical example of medical sensors includes wearable sensing devices that measure different parameters of the body so that a precise inference on health-related problems can be carried out.

- *Neural Sensors.* These acquire neural signals generated in the brain and can be further processed to assist medical staff to make inferences on the current state of the brain or regulate it for overall better focus and attention. The same sensors and appropriate actuators can be used to help patients refocus, manage stress, as well as overall, improve mental well-being.

- Environmental and Chemical Sensors. Environmental sensors are used to sense parameters in the physical environment, e.g pollution levels.

- Chemical sensors can be used in detecting biochemical substances.

- *Radio Frequency Identification (RFID).* RFID readers do not require light of sight to read data from an embedded chip.

2.5.2 Actuators

Actuators are devices, that transform power into some form of useful energy. E.g hydraulic actuators can cause mechanical motion using hydraulic power. Pneumatic actuators use compressed air pressure to cause motion electrical ones to use electrical energy. A combination of any of the three actuators together with appropriate sensors can be used in smart home systems. Typically, actuators can switch on/off the lights, lock/unlock the doors, or, alert users of any intrusions into the home or secluded areas of the home, and can also regulate the room temperatures with the further aid of a thermostat unit.

2.6 Preprocessing

Sensors and actuators in an IoT network will acquire large volumes of data before it is queued for processing. Normally the computing, as well as storage resources, are cloud-based as the cloud offers more storage capacities together with improved data handling, scalability, and flexibility. This may not always suffice for most applications and services for the following reasons.

Lack of reliable and real-time actuation: Latencies associated with communicating at the Cloud level often are long and that does not suit most real-time applications and services. Typically, UDP transport protocols are used to ensure faster communication. However, UDP is an unreliable protocol and in any case, wireless links tend to be comparatively lossy.

Mobility issues: The mobility of the smart devices is mobile hence meaning their constant location changes, makes communication with the Cloud problematic since the network channel conditions will continuously vary as a result.

Scalability: As more devices request services to the same Cloud simultaneously, so would be the greater latency as the available processing power will have to be distributed to all active requests as a result.

Power constraints: Limited power supply capabilities of the devices as a result of them being miniaturized and as such the battery sources are correspondingly small. The devices can therefore only communicate for short durations.

Mobile cloud computing (MCC) was initially proposed as a possible solution to the aforesaid problem, but still, it also suffers latency, signal fading, and power-related problems. Rather of late Fog Computing (which brings cloud resources to the network periphery) which precomputes the data before sending it to the more distant Cloud computing resources was proposed. In short, data is preprocessed at the network edge before relaying it to the Cloud. The two paradigms collaborate to achieve optimal performance in terms of IoT applications and services. A fog node (smart gateway) can be introduced between underlying networks and the cloud to realize fog computing.

Typical Fog computing features are as follows:

Low latency: relatively low turnaround times for submitted processing jobs, i.e. less time to access both storage as well as computing resources on the smart gateways.

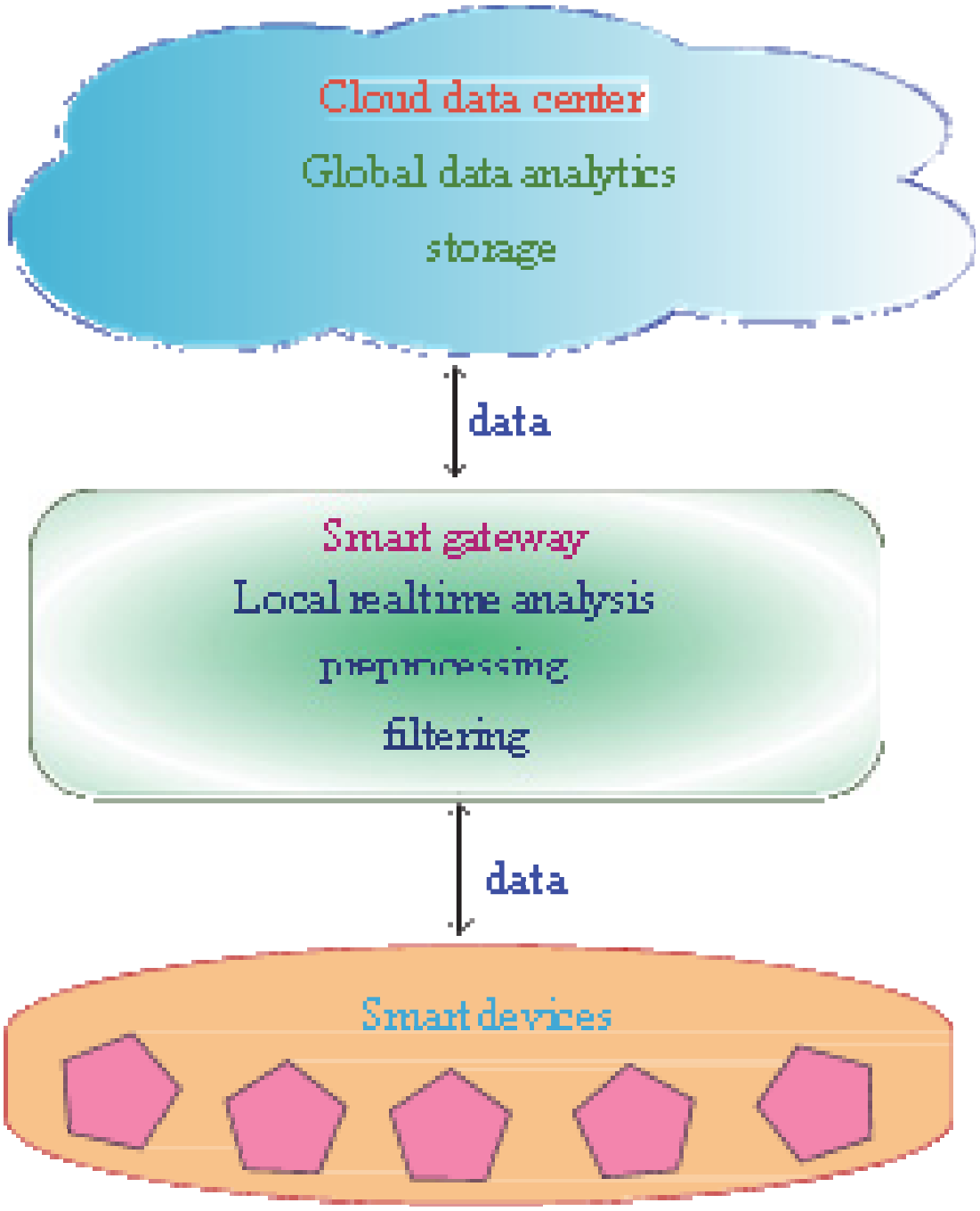


Figure 2.4: Pre-processing at Smart gateways

Improved location awareness: This is attributed to by virtue of the fog being located at the periphery of the network and in that way it is always aware of both the location and contexts of the IoT applications and services.

Distributed nodes: The fog nodes evenly distributed throughout the network are distributed. By nature, a distributed network is more reliable as well as robust as compared to a centralized one. Typical applications and services will enjoy improved latencies as well as turnaround times.

Mobility: Smart devices tend to intercommunicate with smart gateways in proximity hence indirect support for mobility.

Real-time response: Fog nodes support almost real-time responses, unlike the case with cloud computing.

Interaction with the cloud: Fog nodes can selectively communicate the cloud i.e by relaying only the required data to the cloud.

Typical applications of fog include are in Smart vehicle networks and Smart Grids.

Smart vehicular networks: This is an application deployed in the form of smart gateways to regulate the movement of cars and vehicles using sensors.

Smart Grids: A Smart Grid's Automated Smart Metering will regulate demand, supply as well as usage of available grid power to all users. In the process, it will always achieve load balancing. This balancing is typically effected at the edges of the network using smart meters installed in individual households or microgrids connected through smart gateways. The gateways can analyze data related to energy usage and demands and in the process can predict anticipated future demands.

2.7 Communication

A fully-fletched IoT network will accommodate both constrained and unconstrained objects and devices. Their heterogeneous nature also adds complexity. All the devices also have both limited compute and storage capacities. The constrained nature of some or most of the devices and objects poses communication challenges such as:

- Addressing and identification: Each device and object will have to be uniquely physically as well as logically identified. Hence the need for large address space to accommodate the billions of devices connected.
- Low power communication: Low power communication modes would be ideal. However, lots of these low power devices do not have sufficient power during communication modes
- Routing protocols: The efficacy of routing protocols in terms of memory requirements is desirable.

Reliability as well as, high rate communication (link speeds) is also a desirable characteristic of all devices and objects connected to the |IoT

2.7.1 Near Field Communication (NFC)

This is a short-range RFID-based wireless communication standard technology and this facilitates, through mobile devices within proximity of each other to communicate. All adapt types can be exchanged rapidly between any two NFC-enabled devices provided they are in proximity. NFC operational frequency band is around 13.56 MHz, i.e. it is the same as that of RFID. It does operate in two modes namely passive and active. In an active mode, both sender and receiver devices generate magnetic fields, while in the passive mode, only one device generates the field and the other uses load modulation to transfer the data. The passive mode is more energy efficient. NFC can also support duplex communications. All smartphones today are NFC-enabled.

2.8 Wireless Sensor Networks (WSN) Based on IP for Smart Objects

Because of the limited geographical coverage RFID, NFC, Bluetooth, and others were gradually availed. They acquire environment-related data before relaying it to cloud servers via gateway devices for final processing.

2.8.1 IoT Network Protocol Stack

The IETF has developed alternative IP-compatible wireless communication-based protocols for communication between IoT devices. An example is the Internet Protocol for Smart Objects (IPSO) Alliance that has proposed protocols and standards for the layers of the IP stack and an additional adaptation layer to cater for communications between smart objects.

2.8.2 Clouds of Things

This is a platform for rapidly provisioning a set of pooled configurable computing resources through enabling, on-demand network access in IoT-enabled networks, [17, [18]. This is illustrated in Figure 2.5.

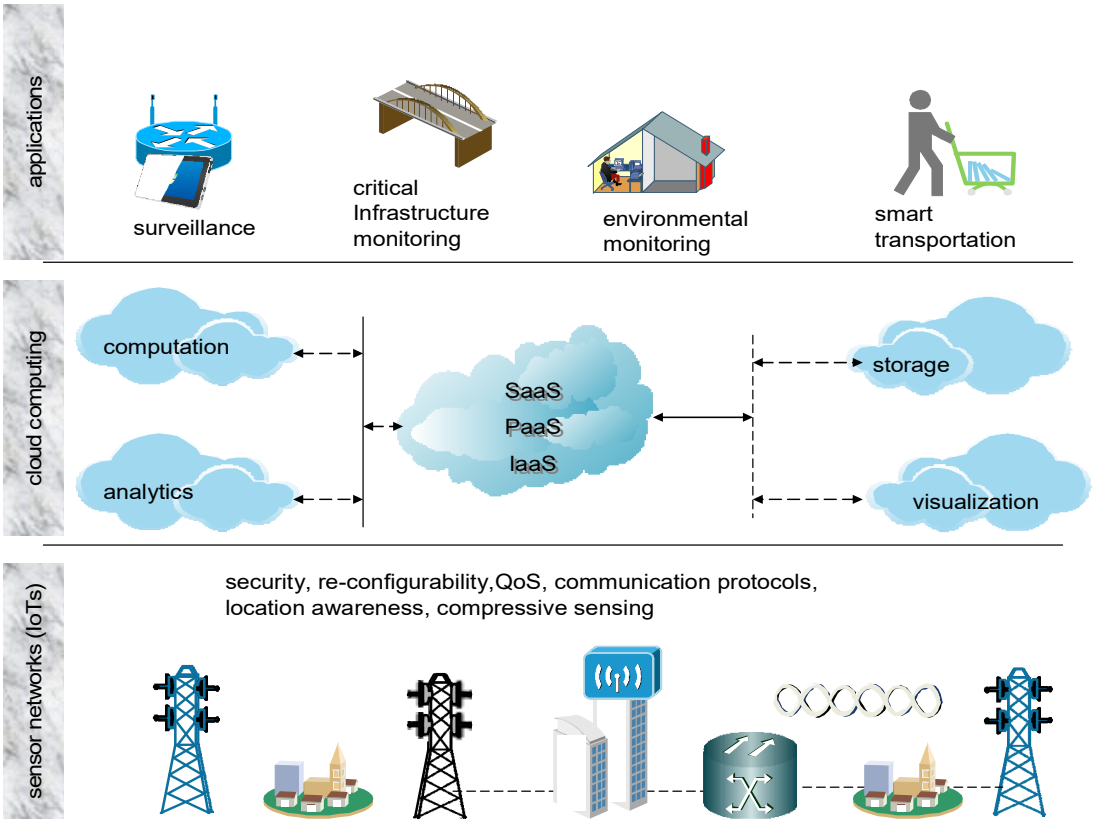


Figure 2.5: Cloud as middleware in IoT paradigm

Typical cloud computing characteristics are: -

- *On-Demand self-service:* i.e, the ability to render users instantaneous access, to computing resources requirements (e.g. CPU time, storage space, network access, etc.) without requiring any human interaction with the provider of those resources.
- *Network Access:* Such requested resources are deliverable through the IoT-enabled network and accessible to several clients as well as client applications with diverse platforms requiring standard protocols and mechanisms to access them.
- *Resource Pooling*: The available resources are pooled together to serve many customers concurrently utilizing various dynamically assigned physical and virtual resources to satisfy customers' QoS expectations. This "multitenancy" model relies on the use of virtualization and in that way, IT resources can be dynamically assigned and reassigned, according to demands.

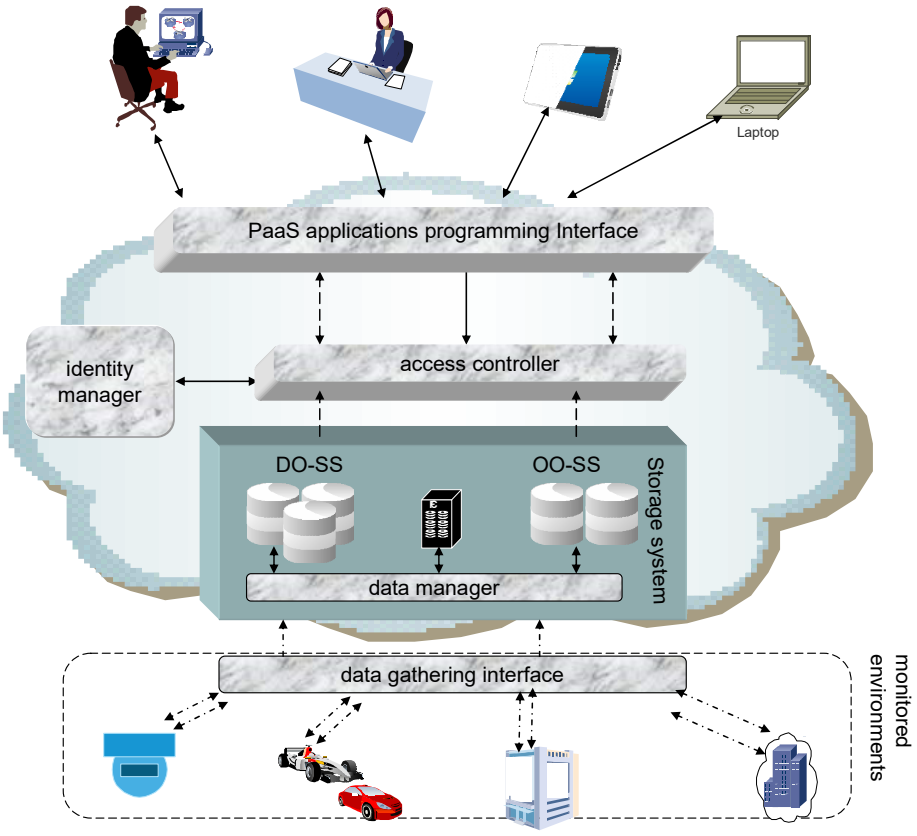


Figure 2.6 Cloud Storage System

- *Rapid Elasticity*: The service provisioned by a cloud provider elastically deployed, assigned, released, or scaled as per demand.
- *Measured Service*: The ability of the cloud service to monitor and measure actual individual usage and charge fairly. In terms of infrastructural deployment within the IoT context, four models exist, and these are [19]:
    - *Private Clouds*: This infrastructure is provisioned to an individual organization so that it restricts access and usage of the services it avails of employees.
    - *Community Cloud*: This is an infrastructure to a community that shares a common goal *Public Cloud*: Such an infrastructure's services are provisioned for open use on a pay-per-use model.
    - *Hybrid Cloud*: In this case, the infrastructure blends two or more distinct infrastructure deployment models.
    - *Inter-Clouds (Cloud Federations)*: This is a relatively newer cloud provisioning model that offers more flexibility, as well as improved reliability and geographic distribution.

Depending on cloud services that are renderable by cloud providers, three service models are specified. These differ on control granted to requested resources by a user as well as, the general functionalities and the architectural layer offered.

- Software-as-a-Service (SaaS): In this case, the users rent out their applications via a service provider.
- *Platform-as-a-Service (PaaS):* This is primarily a development platform that is provisioned to customers to develop their proper applications or services.
- *Infrastructure-as-a-Service (IaaS):* The users are allowed direct usage of the IoT infrastructure. This includes processing, storage, and network resources. In practice, this is implementable through virtualization techniques.

The convergence between Cloud Computing and IoT has led to the "Cloud of Things" or CloudIoT.

With the advent of IoT, storing data locally and temporarily will not be feasible anymore as more storage space would be required. In any case, most of the data would require processing externally (in the Clouds) where there are better, efficient, and more capable computing resources.

Primarily, IoT services are provided as an isolated vertical solution in which a given application and related components are tightly coupled to the specific context of the application. Coagulating and rendering IoT services via the Cloud will ease the delivery and deployment of them by leveraging all the flexibility of Cloud models. In this regard, Cloud computing facilitates application development and makes possible an abstract vision of the IoT systems. IoT can also provide a platform for the Smart Cities services that are envisaged in the next 5-10 years.

2.9 Related Alliances, Organizations, and Standards

As previously emphasized, the IoT networking approach will allow any communication-capable devices and objects to communicate and in the process can be utilized to make collaborative decisions that are beneficial to humanity. However, the underlying diverse technologies of the various devices, objects, and systems impose lots of connectivity challenges and hence introduce the need for specialized standards and communication protocols. In this section, we highlight efforts by several standards, associations, and consortiums to collaborative-

ly develop IoT protocols and standards that are operating at different layers of IoT networking.

2.9.1 Key IoT Related Organizations

Key Organizations related to IoT development and deployment activities include [20]:

- *The European Telecommunications Standards Institute (ETSI)* focuses on connecting "Things" as well as clustering them.
- *The Internet Engineering Task Force (IETF)*: This is the current Internet's leading standards-setting body that has since set up an additional IoT Directorate Group that is spearheading and coordinating related efforts in reviewing specifications for consistency, and monitoring IoT-related matters.
- *The Institute of Electrical and Electronics Engineers (IEEE)* focuses on IoT-related innovations as well as specifications.
- *Object Management Group (OMG)* focuses on Data Distribution Service Portal;
- *The Organization for the Advancement of Structured Information Standards* (OASIS) whose MQTT Technical Committee spearhead IoT related issues;
- *Open Geospatial Consortium (OGC)* focusing on Sensor Web for IoT Standards Working Group;
- *The European Lighthouse Integrated Project addressing IoT Architecture (IoT-A*) focuses on the formulation of a standardized protocol/architectural reference model for the IoT.
- *One_M2M*, which proposes a single or one M2M and hence are also focusing on developing technical specifications for a universally standardized M-2-M Service Layer whose compatibility with various hardware and software enables reliable interconnection of all devices with M2M application servers globally.
- *Open Standards IoT (OSIoT*) whose focus is on developing and promoting free open source standards.
- *Eclipse Paho Project*: This is an organization that focuses on the overall integration of D2D/M2M applications.
- *OpenWSN*: This is a platform as well as a repository for open-source implementations of protocol stacks based on IoT standards.
- *CASAGRAS*: An initiative by Europe, the USA, China, Japan, and Korea that addresses universal standards, concerning RFID and its overall role in realizing an IoT.

2.9.2 Alliances

These include [19], [20]:

- *The AllSeen Alliance*: which is focusing on enabling and spearheading universal adoption of IoT-related devices, systems, and products through an open, universal development framework. The AllSeen Alliance is in the process of merging with the Open Connectivity Foundation (OCF) and the merged consortium will retain the OCF name. Overall the merged Alliance will focus on a codebase of diverse and various modular applications and services that facilitate critical activities such as pairing and discovery of neighboring objects and devices, message routing, and security. The cross-platform nature of the open-source codebase facilitates interoperability among diverse as well as basic objects and systems.

- *IP for Smart Objects Alliance* (IPSO) – The IPSO Alliance is an open, forum comprising several organizations and individuals that promote the value of using the Internet Protocol for the networking of Smart Objects. Its R&D efforts are geared towards achieving IoT interoperability by facilitating data metadata exchanges effortlessly, i.e. this is an approach that eradicates the need for translators. The new approach universally defines all objects and devices, so that each no longer requires predefining nor preregistering. Overall, it emphasizes as well as advocates for IP networked devices in healthcare, energy, consumer and industrial applications.
- *Wi-SUN Alliance*: It promotes the use of IEEE's 802.15.4g based interoperability protocol standard to advance seamless connectivity. Primarily, the Wi-SUN Alliance promotes open industry standards for

1. Wireless Smart Ubiquitous Networks and related applications.
2. Advancement, standardization as well as interoperability of wireless Smart Ubiquitous Networks globally.
3. 3. Other activities include user education, industry outreach, and other support programs as well as lobbying regional regulatory bodies for spectrum allocation for smart grid services.

2.10 Protocols

Broadly, IoT candidate protocols can be categorized as Infrastructural, Identification, Communications & Transport) Service Discovery, Data Protocols, Device Management, and Semantic (security).

2.10.1 Infrastructure Protocols

- *IPv6 over Low Power Wireless Personal Area Networks (6LoWPAN).* It is an adaptation layer protocol for IPv6 over IEEE802.15.4 links.
- *Nano Internet Protocol (NanoIP)* - This is a concept that seeks to bring IP-like networking services to embed with sensor devices, by secluding the TCP/IP overheads.

Discovery Protocols

- *Multicast Domain Name System (mDNS*) - Can resolve and map device names to global IP addresses.
- *Universal Plug and Play (UPnP)* - This category of protocols facilitate self-discovery and interaction capabilities by networked sensors and devices.

2.10.2 Data Protocols

- *MQTT for Sensor Networks (MQTT-SN)*: An open protocol designed specifically for mobile and M2M/D2D applications.
- *Constrained Application Protocol (CoAP):* An application layer protocol for WSN nodes.

2.10.3 Communication / Transport layer

- *IEEE 802.15.4*: This is a standard that specifies the physical layer and media access control for low-rate wireless personal area networks (LR-WPANs).
- *ANT*: A wireless sensor network technology- designed for collection and transfer of sensor data and the integration of remote control systems such as controlling indoor lighting or a television set.
- *LoRaWAN:* Network protocol intended for wireless battery-operated devices.

2.10.3 Semantic

- *SensorML*: It is an approved Open Geospatial Consortium standard. That primarily provides standard models and an XML encoding for describing sensors and measurement processes.

- *Media Types for Sensor Markup Language (SENML*): A simple sensor, such as a temperature sensor, could use this media type in protocols such as HTTP or CoAP to transport the measurements of the sensor or to be configured.

2.10.4 Security

Open Trust Protocol (OTrP) - This protocol essentially is designed to enhance and manage security configurations in Trusted Execution Environments (TEEs). It aims at creating an open universal protocol defining how objects and devices trust each other in a networked environment. It uses the Public Key Infrastructure Architecture (PKI), and certificate authorities, as its basic underlying system. X.509 - Standard for managing digital certificates and public keys.

2.11 Middleware

Incorporating both computing and interconnectivity among the multitudes of devices, objects, and systems connected to the IT network poses a huge task. This is because of the varied requirements of the applications, services as well as the devices, and objects that would support their realization of the IoT platform as there is no standardization. In such a situation, a middleware platform appears to be the only practical solution. The middleware will hide the abstract details of the devices and objects to be interconnected by playing a software bridging role between the devices and services. It furnishes the key functionalities to the vendors.

Typical issues include:

- *Interoperability and programming abstractions*: Middleware has to deal with three main issues, namely, network operability, semantic, and syntactic compatibilities. Network interoperability deals with heterogeneous interface protocols for communication between devices. The middleware has to attempt to insulate the various applications from the complexities and intricacies of different protocols. It should ensure Syntactic interoperability among applications that otherwise are based on different

structures, formats, and encoding formats of data. It also has to deal with semantics, ie abstracting the meaning of data within a particular domain.

- *Device discovery and management*: The middleware must provide capabilities for devices to be aware of peers in the neighborhood.

- *Scalability*: The middleware must be adaptable to IoT when the scales up and thus readily provide any necessary changes.

- *Big data and analytics*: Detailing and refining the analysis of the acquired data.

- *Security and privacy*: The middleware ought to ensure security and privacy to users who make use of IoT applications and services.

- *Cloud services*: The IoT middleware should have the capability of seamlessly running on different types of cloud domains and at the same time enabling users to leverage the cloud to get improved insights from the data acquired by the sensors.

- *Context detection*: The data acquired from sensors must be used to extrapolate the context by applying appropriate algorithms. The context can then be used for providing improved services to users.

Quite a few middleware solutions have been proposed. These can be categorized into the following types: event-based, Service-oriented, Database oriented, Semantic, and Application-specific. Popular examples of IoT middleware include FiWare and OpenIoT.

2.12 IoT Services and Applications

There are multitudes of applications and services that can utilize the IoT communication platform. In this subsection, we briefly overview some of them without necessarily distinguishing between applications versus services. Four broad categories of IoT applications and services are information, aggregation, ubiquitous, collaborative-aware, and identity-related.

2.12.1 Identity-Related Services

This service has two categories which are active and passive. Either can serve individuals or big corporations. This category service provides two components, the appliances with identification identifiers like IP addresses or RFID tags and the reader which reads the identity of

other appliances. During this identification process, the reader asks permission from the name-resolution server to access detailed data about certain devices [10].

2.12.2 Information Aggregation Services

Information aggregation service is for the transmission of data in IoT services. Acquires information from sensors processes it, transmits and reports that collected information through IoT to the applications. In this process, there is no implementation of any communication channel required for them to work together. This service uses different types of sensors with access to gateways and network devices to share data through the common service to the application. RFID tags are used during communication to be aware of the device's identity, to collect data from sensors Zigbee network is used then gateway devices are responsible for relaying data to the application under a similar service.

2.12.3 Collaborative Aware Services

Collaborative-Aware Services uses the collected data to decide, then takes action based on the decision taken. These services retrieve data and send back responses to those devices to perform actions.

2.12.4 Ubiquitous Services

This service is always the essence of IoT and everything. For the IoT to be able to deliver this service it needs to meet all the protocol requirements in the technologies and merge every part of the network. [10].

2.12.5 Example Applications and Services

- Smart Mobility- The service ensures easier and safer transportation methods for metropolitan residents, communicators as well as travelers. It generally involves network security, vehicle-to-vehicle communication security, and vehicle-to-infrastructure communication. Smart sensors guarantee eco-friendly driving. Another objective is the automatic monitoring and identifying the critical systems and dangers in road warnings.
- Smart City and Transport - These are applications designed to manage daily traffic in metropolitan areas using sensors and intelligent information processing systems. The main aim of intelligent transport systems is to minimize traffic congestion, ensure easy

and hassle-free parking, and avoid accidents by properly routing traffic and spotting drunk drivers. The overall service is achieved by a collaboration of several applications such as; traffic surveillance and management, traffic congestion detection, intelligent parking management, smart traffic lights, and accident detection.

- Smart Home - This is a combination of applications meant to provide safety in our homes. It requires the installation of a home IP network that will facilitate and afford comfort, convenience, assisted living, and environmental monitoring. Sensors can be installed in the home to acquire environmental data like temperature, lighting, humidity, noise, and atmospheric pressure. The smart-home application can in turn use the data to make the necessary controls and regulations in the home.

- Smart Health - These facilitate remote health monitoring. Appropriate sensor devices can be fitted on our bodies and they, in turn, obtain the necessary body data before relaying it to medical specialists.
- Smart Water Systems - Smart water meters can be installed to measure the degree of water inflow and outflow and to identify possible leaks. Smart water metering systems are also used in conjunction with data from weather satellites and river water sensors. They can also help us predict flooding.

2.13 Security Features of IoT

Confidentiality: This security feature ensures that the data is available only to authorized users throughout the process, and it cannot be eavesdropped on or interfered with by unauthorized users. Confidentiality is an important security principle in IoTs because a lot of measurement devices (RFID, sensors, etc.) are integrating with IoT. This means that it is critical to ensure that the data collected by a measurement device will not reveal secure information to its neighboring devices. The enhanced techniques, including secure key management mechanisms, and others should be developed and used to achieve better confidentiality [22].

Integrity: Integrity ensures that data (transmitting/received) cannot be tampered with by any interference during the data delivery in communication networks, in the end providing accurate data for authorized users. If IoT applications receive forged (Tampered) data inaccurate operation status can be estimated, and wrong feedback commands can be made, which could then disrupt the operation of IoT applications this means that Integrity is very important for

IoT. For acceptable integrity to be achieved, enhanced secure data integrity mechanisms (false data filtering schemes, etc.) should be developed and applied.

Availability: Availability is also an important security principle. Whenever data and devices are requested, Availability ensures that the data and devices are available for the authorized users and services. IoT services are commonly requested in a real-time fashion, and services cannot be scheduled and provided if the requested data cannot be delivered promptly. One of the most serious threats in availability is the denial-of-service (DoS) attack, and enhanced techniques that are secure and efficient routing protocols should be studied and applied to ensure availability in IoT [23].

Identification and Authentication: Identification ensures that the unauthorized devices or applications cannot be connected to IoT, and authentication ensures that the data delivered in networks are real, and the devices or applications that request the data are also real.

Privacy: Privacy ensures that data will only be controlled by the corresponding user and that no other user can access or process the data. Not like confidentiality, which makes sure that data is encrypted without being eavesdropped on and interfered with by nonauthorized users, it ensures that the user can only have some specific controls based on received data and cannot conclude other valuable information from the received data [23], [24]. Privacy is taken as one of the important security principles due to many devices, services, and people sharing the same communication network in IoT.

Trust: Trust can ensure that security and privacy objectives are achieved during the interactions among different objects, different IoT layers, and different applications. The objectives of trust in IoT can be divided into trust between each IoT layer, trust between devices and applications [13]. Security and privacy can be enforced with trust. Trust management systems should be designed to implement these trust objectives in IoT.

2.13.1 Security

The security challenges that are available in each layer of IoT architecture are presented in detail. The service layer is responsible for extracting the functionality of data services in the network layer and the application layer. The security challenges in this service layer can be attributed to challenges in the network and the application layers. Only security challenges in the perception layer, the network layer, and the application layer are presented.

Perception Layer: The purpose of the perception layer in IoT is to collect data. The security challenges in this layer are on forging the collected data and destroying perception devices, which are explained in detail below:

- *Node capture attacks:* In this attack, the opponent can capture and control the device in IoT by physically replacing the device or tampering with the hardware of the device [25]. If the device is compromised by the node capture attack, the valuable is exposed to the opponent. The opponent can also be able to copy the valuable information associated with the captured node to a malicious node, and then fake the malicious node as an authorized node to connect to the IoT network or system. The node capture attack is represented as the node replication attack. This attack can incur a serious impact on the network. For this attack to be defended, effective schemes to monitor and detect malicious nodes need to be studied.
- *Malicious code injection attacks:* The opponent can control a device in IoT by injecting malicious code into the memory of the device, which is denoted as the malicious code injection attack. Then the injected malicious code not only can it perform specific functions, but it can also grant the opponent access into the IoT system, or even gain full control of the IoT system. For the malicious code injection attack to be defended, effective code authentication schemes need to be designed and integrated into IoT [26], [27].
- *False data injection attacks:* Besides, from the captured device in IoT, the opponent can insert false data where normal data has to be measured by the captured device and transmit the false data to IoT applications [24]. After receiving the false data, the IoT applications can return inaccurate feedback commands which further affects the effectiveness of IoT applications and networks. For such a malicious attack to be defended, the techniques (false data filtering schemes, etc.) which can efficiently detect and drop the false data before the data is received by the IoT applications, need to be designed.
- *Replay attacks (or freshness attacks):* In the IoT, the opponent can use a malicious node or device to transmit to the destination host with legitimate identification information, which has been received by the destination host, to make the malicious node or device obtain the trust of IoT [26], [23]. The replay attack is commonly launched in the authentication process to destroy the validity of the certification. To alleviate the replay attack, this technique of secure time stamp schemes should be designed and developed in IoT.

- *Cryptanalysis attacks and side-channel attacks:* The cryptanalysis attack can use the attained ciphertext or plaintext to get the encryption key that is being used in the encryption algorithm [17] [22]. The efficiency of cryptanalysis attacks is very low. For the improvement of this efficiency, the new attacks namely the side-channel attacks can be introduced by the opponent. Like the side-channel attack investigated in IoT [17] [37], the opponent could organize some techniques on the encryption devices in IoT to get access to the encryption key, which is used in IoT for encrypting data and decrypting data. The typical side-channel attacks are the timing attack, in which the opponent can get the encryption key by analyzing the time information required to execute the encryption algorithm. For the side-channel attack to be investigated, efficient and secure encryption algorithms and key management schemes need to be developed in IoT [28].
- *Eavesdropping and interference:* Most devices in IoT communicate through wireless networks, weakness lies in the fact that information delivered in wireless links can be eavesdropped on by nonauthorized users. For this eavesdropping to be dealt with, secure encryption algorithms and key management schemes are required. The opponent can send noise data or signals to interfere with the information delivered in wireless links. To make sure of the accuracy and timely delivery of data, effective secure noise filtering schemes are required to filter the noise data and restore the original data.
- *Sleep deprivation attacks:* Most devices have low power ability in IoT, to extend the life cycle the devices are programmed to follow a sleep routine to reduce power consumption. The sleep deficiency attack can break the programmed sleep routines and keep the device awake all the time until they are shut down. For the life cycle of these devices and nodes to be extended, the energy harvest scheme can be a possible solution, in which devices can harvest energy from the external environment like solar. The other techniques, like the secured duty-cycle mechanism to alleviate the sleep deprivation attack, need to be studied in IoT.

Network Layer: The main purpose of the network layer in IoT is to transmit the collected data, the security challenges in this layer focus on the impact of the availability of network resources. Most devices in IoT are connected to IoT networks through wireless communication links. Therefore, most security challenges in this layer are related to the wireless networks in IoT.

- *DoS attacks:* This attack can consume all the available resources in IoT. It can consume them by attacking network protocols or bombarding the IoT network with high traffic, rendering the services of IoT systems unavailable [20]. The DoS attack is one of the most common attacks in IoT and is one of the attacks which can result in the services of IoT systems being unavailable. Even though this attack can be generated by attack schemes, including, UDP flood, Ping of Death, SYN flood, TearDrop, and Land Attack. To defend the IoT against this attack, the attacking schemes need to be carefully investigated first, and efficient defensive schemes to mitigate attacks need to be developed to secure IoT systems.
- *Spoofing attacks:* The spoofing attacks allow the opponent to gain full access to the IoT system and send malicious data into the system. Examples of spoofing attacks in IoT include IP spoofing and RFID spoofing]. Checking from the IP spoofing attack, the opponent can spoof and record the valid IP address of other authorized devices in the IoT. Then access the IoT system to send malicious data with the obtained valid IP address making malicious data appear to be valid. From the RFID spoofing attack, the opponent can spoof and record the information of a valid RFID tag and send malicious data with this valid tag ID to the IoT system. The possible solutions to defend against the spoofing attack can be Secure trust management, identification, and authentication.
- *Sinkhole attacks:* In this attack, a compromised device claims exceptional capabilities of computation, communication, and power, such that more neighboring devices will select the compromised device as the forwarding node in the data routing process because of the appealing capabilities. Therefore, the compromised device can increase the amount of data obtained before it is delivered to the IoT system. The sinkhole attack not only can break the confidentiality of delivered data but also can be a fundamental step to launch additional attacks. Techniques such as secure multiple routing protocols need to be studied and applied to defend against sinkhole attacks.

Application Layer: The application layer's most key role is to support services requested by users. The challenges in the application layer focus on software attacks. Some possible challenges in the application layer of IoT are:

- *Phishing attack:* The opponent can obtain the confidential data of users, such as passwords and identification, by spoofing the authentication credentials of users through infected e-mails and phishing websites. Secure authorization access, authentication, and identification can mitigate phishing attacks. The most efficient way is for the users

themselves to always be attentive while surfing online. This is an issue because most of the devices in IoT are machines, which may lack such intelligence.

- *Malicious virus/worm:* A malicious virus is another challenge to IoT applications. The opponent can infect the IoT applications with malicious self-propagation attacks and then obtain or tamper with confidential data. To combat malicious virus/worm attacks in IoT applications a reliable firewall, virus detection, and other defensive mechanisms need to be deployed.
- *Malicious scripts:* This represents the scripts that are added to the software, modified in software, and deleted from software to destroy the system functions of IoT. As all the IoT applications are connected to the Internet, the opponent can easily fool the customers into running malicious scripts like the java attack applets or the active-x scripts, when requesting services through the Internet. The malicious scripts can pose leakage of confidential data and a complete system shut down. To be protected against malicious scripts there are some effective script detection techniques in the IoT systems which need to be deployed like honeypot techniques, static code detection, and dynamic action detection.

2.13.2 Privacy

To preserve privacy on the original data or to hide the sensitive parts, data perturbation techniques are used. During transmission and receiving of data, noise is always an issue, to minimize these issues, anonymous techniques are used. There are noise addition techniques that are used to add noise to the original information for the message to not be readable to hackers. Data sampling, noise random, data swapping and differential privacy are the four groups of this technique implemented to add noise on the message to be unreadable. To hide the data owner's identity by removing any unambiguous identifier making data unclear is the other technique used called anonymous protection. K-anonymity, I diversity, and t-closeness are the methods for privacy-preserving. The data restriction technique encrypts the inputs and blocks access to limit data usage. This method controls access to data and uses cryptography-based techniques.

The technique that is effective in ensuring data sharing is called access control. The data owners have the privilege to choose who can get access to see their data and how others manipulate their shared data. Cryptographic protection techniques are used mostly when preserving the privacy of the shared data, the secure multiparty computation, a cryptographic method that

uses keys to encrypt and decrypt data are asymmetric/symmetric encryption, public key infrastructure.

To ensure privacy in line of communication in IoT all of the massive data collected and used should go through the following three steps:

i) Data collection.

ii) Data aggregation.

iii) Data mining and analytics.

The data collection is approved to sense and collect the status data of objects in IoT. The data aggregation integrates the amount of the related data into a piece of comprehensive information, data mining, analytics extract, and the potential value of integrated comprehensive information for special applications in IoT [28]. Even though data aggregation, data collection, and data mining, and analytics can provide several services to our lives. As privacy is a new challenge in IoT, it can lead to property loss, and it can even compromise human safety [25], [28]. For instance, in the smart grid, if the opponent obtains the private data of the energy consumption of customers, it can conclude the time when users are at home or out of home, then it can conduct theft or other damage to users with a probability. The privacy-preserving mechanisms need to be put in the system to ensure private data not be leaked to the adversary in IoT. Based on different data processing steps, privacy-preserving mechanisms can be divided into three categories: Privacy preservation in data collection. Privacy preservation in data aggregation and privacy preservation in data mining and analytics. Privacy in data collection, data analytics, and data mining can be greatly preserved by various techniques like data encryption and key management. Most of the existing efforts on privacy preservation in IoT focuses on data privacy in data aggregation.

2.14 Summary Chapter Conclusions

IoT-enabled networks have the potential to transform our everyday lives in various sectors such as health, transport education, entertainment, and general interaction. This is because it can support a wide range of services and applications. However, as the number of interconnected devices surges, coupling with a demand for commensurate data speeds, we inherently are faced with security and privacy challenges. We also face limitations in the degrees of freedom to develop standard protocols for the IoT network that will ensure harmonious opera-

tion(s). The chapter overviewed an overall architecture for IoT, standards as well as potential new threats for the security, privacy, and trust (SPT) at different levels of architecture. The chapter also highlights the problem of resource constraints faced by most devices. It is noted that most of the devices low powered and hence not able to communicate directly over long ranges. Rather they can only do so over short ranges, hence the next chapter will explore, D2D communication paradigm as it is viewed as ideal for resources-constrained environments.

3. D2D Communications

3.1 Introduction

Device-to-device (D2D) communication facilitates as well as supports direct communication between devices in proximity. It is a feature incorporated in 5G IoT and GSM cellular networks. It ensures interoperability between public network infrastructures, critical public safety networks, and other ubiquitous networks such as the current LTE. Its goal in supporting proximity-based communications is to improve spectrum utilization, spectrum efficiency, end-to-end throughput, and energy efficiency, and at the same time facilitate novel applications and services. D2D-communication compatible devices and objects are potentially a fail-safe back-up infrastructure for critical mission networks should the public cellular networks are suddenly malfunction or shut down totally. Issues have arisen as to the sharing of available spectrum between cellular and D2D communications. Notably, it is not clear whether D2D devices should utilize OFDM resources or opportunistically access the spectrum resources occupied by cellular mobile. Further complexity is in the new design flexibility of D2D mode selection that enables D2D communication-enabled devices to switch between conventional cellular and direct and communications [29].

3.2 D2D Communications

D2D communications is a feature that facilitates communication among devices and objects (also referred to as user equipment (UE)) in the presence or -non-presence of network infrastructure such as the public switched network (PSN) or similar networks. Typically there is no need for a network access point (AP) such as a base station (BS). They rely mostly on their proximity to communicate directly in what is referred to as direct linkage. Typically this technology is integrated with the next generation IoT/LTE network infrastructures. In operational terms, the standard network would authorize the devices to communicate directly without its involvement. In this case, it would sanction a service provider to determine the direct path routing in the network. When the network is down, the peer devices can communicate directly.

- Overall advantages of D2D communications are summarized as follows: provisioning of ultra-low delay communication, thus offering more reliable linkages between peer devices.

- Direct communication between devices aids in the improvement of network capacity for networks such as GSM extension of network coverage. This is the case when a particular device is at a cell edge and hence the network is fading. It then directly links with a nearby peer device that is in a good reception area and thus consequently network coverage has been extended.

- It can aid network resilience in that should one pair of devices fail to communicate, due to an intermediary BS failure, the affected devices can still invoke direct linkages with neighboring peers and still communicate. In that was service is not interrupted.

- D2D communications provides flexibility in offloading traffic from devices completely off the network. In that way, both spectral and energy efficiencies are enhanced. As the energy cost per bit is drastically reduced.

- D2D communications is relatively resilient and non-susceptible to multi-user interference as it typically utilizes wave frequencies that allow multiple D2D links to operate simultaneously and in the process make it very robust to multi-user interference.

- By nature, it offers better privacy and security as data are never stored at a centralized node (switch)

A few notable D2D communications disadvantages is as follows:

- The devices themselves are utilizing periodic clock signals (broadcasts). Should the synchronization fail, then devices may not be able to communicate.
- The peer discovery algorithm used in D2D communications can be quite complex in a clustered cell network since getting cooperation from adjacent BSs may be problematic.
- Interference management issues are quite prominent in D2D communications. Typically, inband D2D communication, cellular link, and D2D link interfere with each other. In Outbound D2D communications, D2D links interfere with each other as well as with other devices using the same band. UEs transmit low power to reduce the interference but at the cost of QoS (Quality of Service).
- Billing of users is near impossible because of the lack of centralized coordination.

- Security risks such as eavesdropping, IP spoofing, denial of service, malware attacks are quite prominent in D2D communications [28], [30].

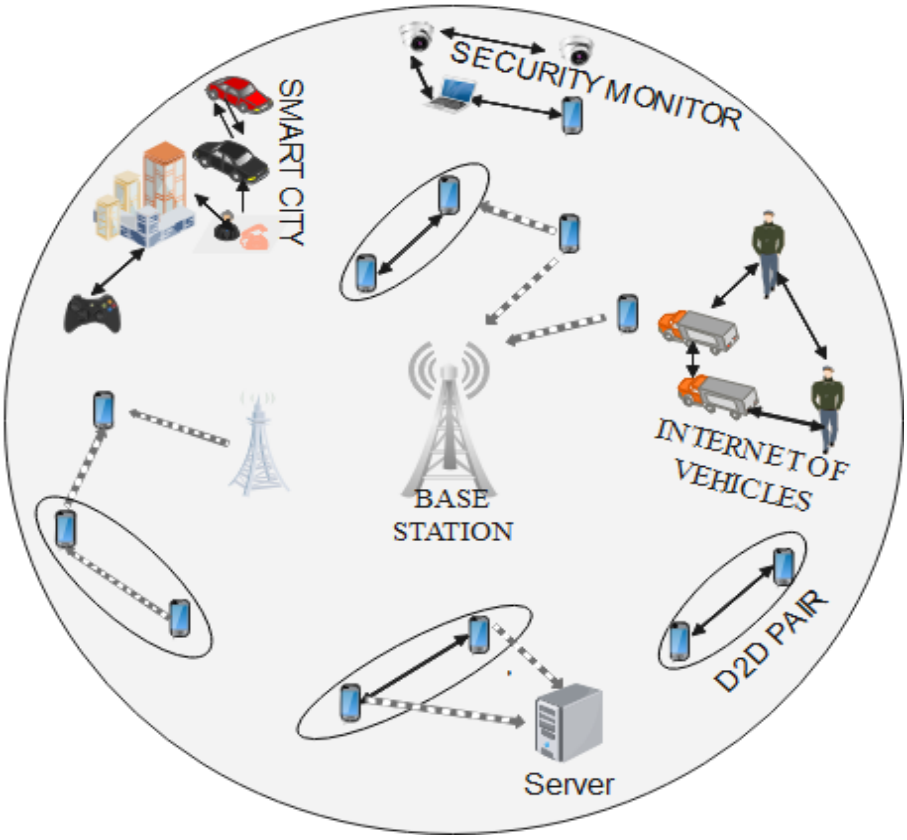


Figure 3.1: D2D Communications Scenario

3.3 Overview Classification of D2D communications

D2D communications is primarily categorized as; In-band D2D and Out-band D2D. With In-band D2D, the same licensed spectrum is used for both cellular communication and D2D communication. Outbound D2D, D2D uses an unlicensed spectrum where cellular communication does not take place.

In terms of communication, there exist two types of communications namely single-hop and multi-hop. In single-hop communication, transmitting UE and receiving UE connect directly. In multi-hop communication, intermediate UEs act as relays either between "BS and UE" or between "two UEs".

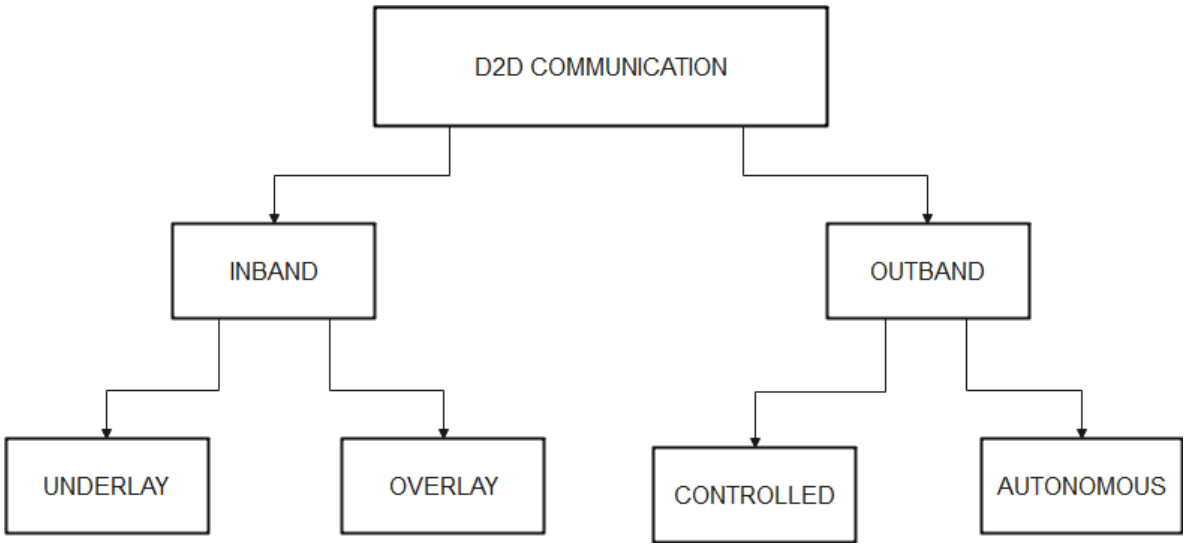


Figure 3.2: Types of D2D communications

3.4 Security Architecture

With the rapid development as well as deployment of wireless mobile communication technologies, the 3GPP likewise is proposing new 5G standards to enable a smooth migration from the current LTE systems to the next generation mobile communication system (5GS). Accordingly, a security architecture for D2D communications was proposed in [18]. In comparison to present LTE/LTE-A systems, the new security architecture incorporates various entities and functions. Examples include Unstructured Data Storage Function (UDSF), Structured Data Storage Network Function (SDSF), Network Exposure Function (NEF), Network Repository Function (NRF), Policy Control Function (PCF), Unified Data Management (UDM), and Application Functions (AF) [18]. An example of security architecture is illustrated in Figure 3.3. In the same figure, two functional entities namely; the ProSe Function and ProSe App Server, and five reference points PC1, PC2, PC3, PC4, and PC5 in are shown. ProSe Function has several functional modules that are deployed in the core network. It also interacts with the Evolved Packet Core (EPC) through reference point PC4, with the ProSe application server via PC2, and with D2D devices (UE) via PC3. There are main functionalities of ProSe Function, which include the service configurations for UE, the UE discovery, security, and trust management.

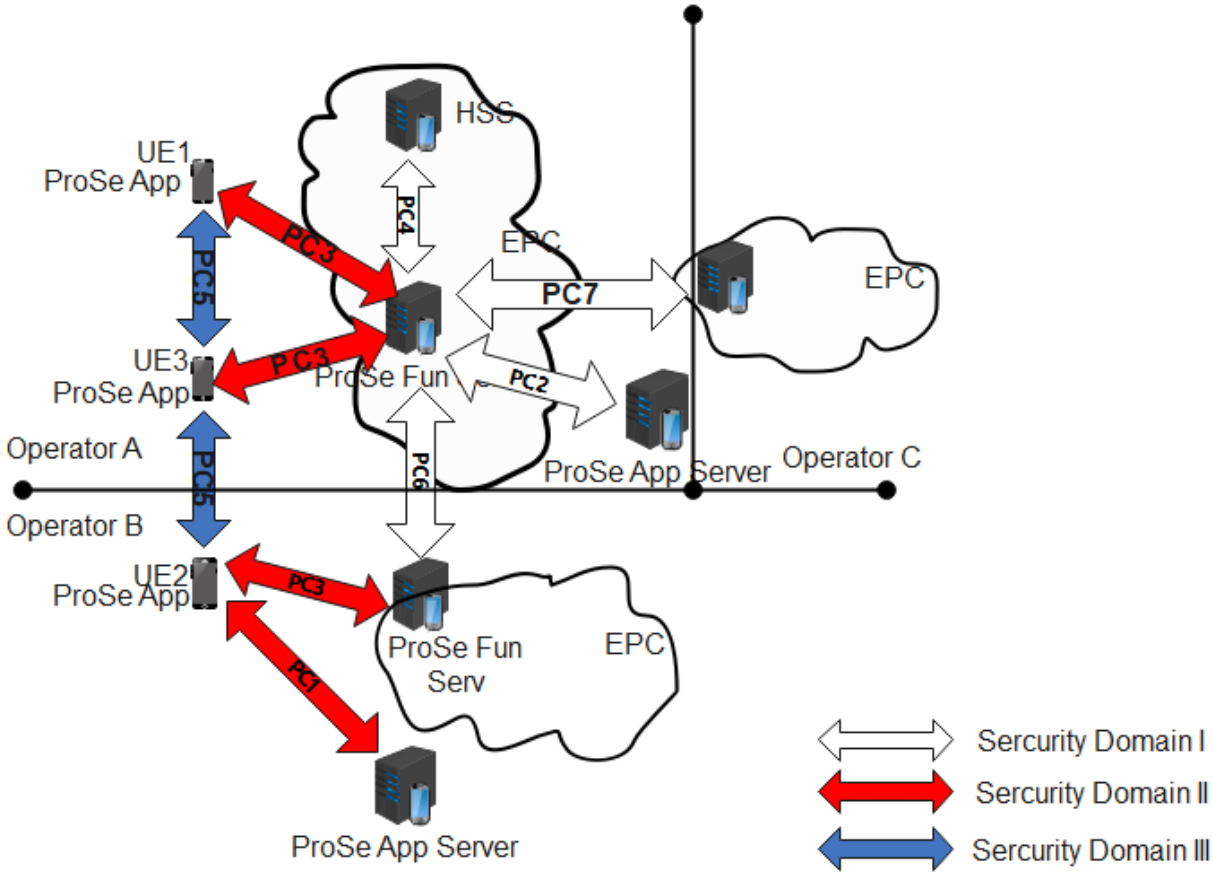


Figure 3.3: Security Architecture

This server is an application server, which provides D2D related application services for users. It can communicate with ProSe Apps like the mobile app installed in UEs for service provisioning through PC1. The UEs can communicate directly with each other through the reference point PC5, an entirely new radio interface introduced by D2D communications. The 3GPP has defined five LTE system security levels:

- Network access security
- Network domain security
- User domain security
- Application domain security
- Visibility and configurability of security.

3.5 Chapter Summary

The chapter overviewed the advantages and disadvantages of D2D communications. It is noted that it is key in facilitating direct communication between devices in proximity. It also aids in ensuring interoperability between public network infrastructures, critical public safety

networks, and other ubiquitous networks such as the current 5G/LTE. Its goal in supporting proximity-based communications is to improve spectrum utilization, spectrum efficiency, end-to-end throughput, and energy efficiency, and at the same time facilitate novel applications and services.

4. D2D Communications Based Authentication Protocols

4.1 Introduction

As cited before D2D communications based technologies focus on facilitating direct linkage and communications among any communicating capable devices, without the mediation of the base traditional transport network infrastructure. In this regard, essential security objectives required for the cohesive interaction among the D2D communicating devices would include, primitives such as the preservation of integrity, confidentiality, robustness, resilience, and availability in case of any intentional or unintended intrusion attacks. This chapter will explore a few protocols in terms of their security capabilities as well as computational complexities in terms of overheads. We will also seek to investigate the energy efficiency of the selected protocols overall.

As such, the focus (objectives) of the chapter is on exploring group authentication-based D2D communications-related protocols. The motivation for exploring such protocols is that most would-be services in the IoT arena are most likely to perform better when their associated devices function collaboratively. This would be the case in e-health, weather monitoring, environmental disaster monitoring, surveillance, and security, etc. In all these cases, the associated devices can be rather organized in groups to address and deliver a common objective much more effectively as well as efficiently.

As can be expected, in the immediate and longtime future, D2D communications is most likely to dominate in GSM wireless networks where device-to-device communication is quite prevalent. Its ultimate goals will be among things to improve the overall service performances of the current 5G/LTE GSM cellular network infrastructure and that of Future Generation cellular networks communications. The communications can be categorized into four category types namely [35]:

- *Direct communication between devices (Direct D2D) with controlled link establishment by the device.* With this category type of communications, associated devices can link directly with each other and the establishment and control are made entirely by the devices involved in the communication.
- *Device relaying with controlled link establishment from the operator*. With this category type, the traditional telephony infrastructure acts as an intermediary in facilitating com-

munication between devices that are both located in areas of weakness. Typically such devices communicate with traditional network infrastructure through direct connection with other D2D devices that can relay their information. The communication establishment and control are made by the base station (BS).

- *Direct communication between devices with controlled link establishment by the operator*. In this case, the devices are capable of directly communicating with each other and the communication management (establishment and session control) is maintained by the BS.
- *Device relaying with controlled link establishment from the device.* The devices are located in areas of weak signal strength. In this case, they can link once again with the traditional network through other D2D devices assisting them by way of relaying their information or data In this type of communication, however, the establishment and control are retained by the devices themselves.

4.2 Key Agreement Fundamentals Overview

In this chapter, we will discuss a few key agreement fundamentals based on Bilinear Pairing, Shamir Secret, Aggregated Signatures, as well as Elliptic Curves Diffie-Hellman (ECDH).

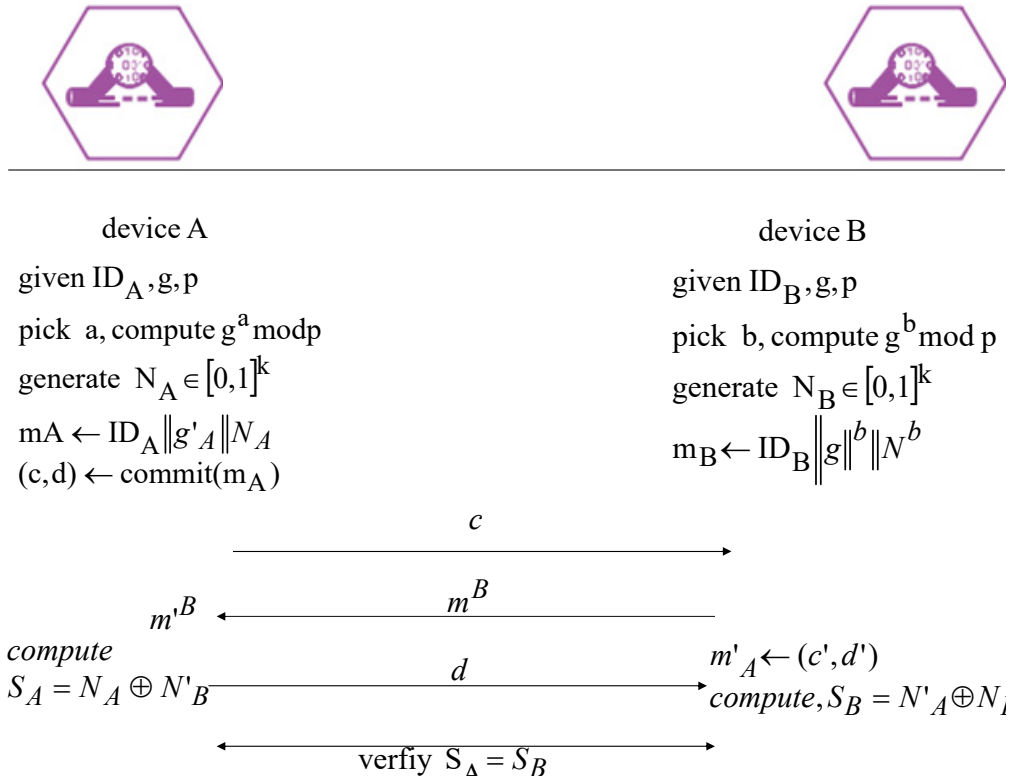


Figure 4.1: Principles of a secure exchange scheme

This is because they form the basis of the group-based authentication we will subsequently review in the same chapter.

4.2.1 Bilinear Pairing

This is a group authentication algorithm that verifies key primitives based on the manipulation of critical parameters [36]. It can be summarily explained in steps:

#1: Initially, a selected prime number *p*, G1 a cumulative group, and GT a product group of order *p* are generated.

#2: A bilinear pairing on (G1, GT) is generated considering the following mapping:

$$e: G_1 \times G_2 \Rightarrow G_T \tag{4.1}$$

The bilinear pairing is subject to the following:

Bilinearity:

For all $R,S,T \in G_1, \hat{e}(R+S,T) = \hat{e}(R,T)\hat{e}(s,T)$ and $\hat{e}(R,S+T) = \hat{e}(R,S)\hat{e}(R,T)$ (4,2)

Non-degeneracy:

$$\hat{e}(P,P) \neq 1 \tag{4,3}$$

Computability:

The value ê is calculated as follows:

$$\hat{e}S,\infty) = 1 \text{ and } \hat{e}\infty,S) = 1 \tag{4,4}$$

$$\hat{e}(S,-T) = \hat{e}(-S,T)^{ab} = \hat{e}(s,T)^{-1} \tag{4,5}$$

$$\hat{e}(aS,bT) = \hat{e}(S,T)^{ab} \text{ , for all } a,b \in \mathbf{z} \tag{4,6}$$

$$\hat{e}(s,T) = \hat{e}(T,S) \tag{4,7}$$

If $\hat{e}(S,R) = 1$, for all $R \in G_1$, then $S = \infty$

Note that the bilinear pairing operation with other schemes to enhance robustness.

$$\hat{e}(S,T)^{RaRbP} \tag{4,8}$$

4.2.2 Shamir Secret

A Group authentication scheme, Shamir's secret was proposed in [37] and among other capabilities, allows entities to mutually authenticate as well as to authenticate as a group. Each entity sends to the other entities in the group its share of the secret. In the $(_{k,n)}$ threshold scheme [36],[37], a given secret D is partitioned into n pieces D_1, $D_2,\ldots,D_n$ and only with at least k pieces, the secret $D\,D$ can be rebuilt. Besides, the secret only can be restored if the pieces are legit.

The advantage of using Shamir's Secret in authentication protocols is that it is fast. Just one verification is necessary to authenticate the whole group of devices. The devices only are authenticated if all devices have proven to have a legit share of the secret. Consequently, a disadvantage is in the impossibility of discovering which device is the intruder. However, Shamir's Secret is used in many areas nowadays, such as image compression, cryptography algorithms, and authentication protocols.

4.2.3 Aggregated Signatures

This is a scheme that was proposed in [38] in which a digitized signature that supports aggregation is utilized. All the digitized signatures from a Group comprising n members are aggregated into a single compacted signature. This short signature will be used by a verifier as an authentication confirmation In general Aggregated signatures provide rapid authentication hence suitable for group authentication protocols This will facilitate a single unique authentication procedure.

Given two Groups, G_1 and G_2, each with its generator g_1 and g_2 respectively, and a bilinear map e.

The bilinear aggregation on these two Groups becomes:

$$G_1 \times G_2 \Rightarrow GT \tag{4,9}$$

The detailed procedure is as follows:

Initially, each member calculates its key;

$$v_i \leftarrow g_1^x \tag{4,10}$$

where x denotes an arbitrary random number chosen by the member.

This is followed by each member computing the hash of a message **M** and corresponding signature σ_i as follows:

$$h_i = H(\mathbf{M}) \tag{4,11}$$

$$\sigma_i = h_i^x \tag{4,12}$$

The aggregation of the digitized signatures is as follows:

$$\sigma = \prod_{i=1}^{k} \sigma_i \tag{4,13}$$

Ultimately the verification of the aggregated signatures is accomplished using bilinear pairing;

$$e(g_i, \sigma) = \prod_{i=1}^{k} e(v_i, h_i) \tag{4,14}$$

4.2.4 Elliptic Curves Diffie-Hellman (ECDH)

This is a key agreement and authentication scheme in which two entities, each with its own elliptic-curve public-private key pair, are accorded the freedom to create a secured link over an otherwise insecure channel. [39], [40].

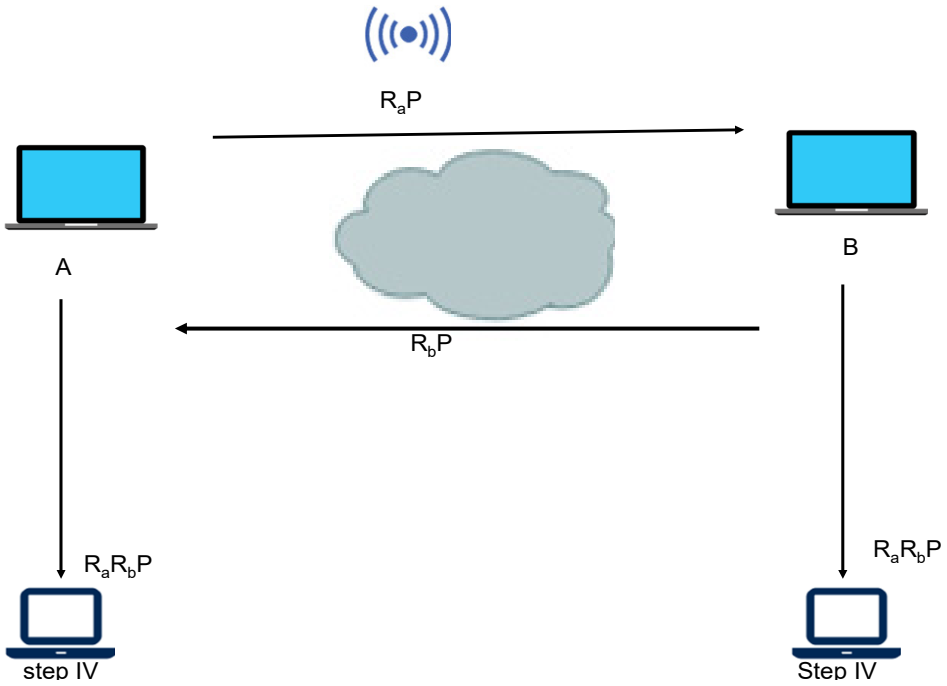


Figure 4.2: An ECDH session example

It is essentially a slight variation of the traditional elliptic-curve cryptography-based Diffie–Hellman algorithm. The authentication procedure can be best elaborated by way of an example as follows:

Given two parties A and B, the ECDH exchange process will proceed as follows:

1. Basic system parameters are initialized. These include a finite prime number p, an elliptic curve E; the field magnitude F_p; and a randomly chosen point P on the inscribed curve (E).

2. The parties each choose a random number, $R_a \rightarrow A$, and $R_b \rightarrow B$ before executing a multiplicative operation over the elliptic curve RaP and RbP.
3. They exchange the operation results in the previous steps as follows:
   $RaP \rightarrow B$ and $RaB \rightarrow A$. (4,15)
4. Each party computes $RaRbI$ before bestowing it as the common shared secret between them. Subsequently, they use this key for any data exchange.

The steps outlined are summarized in Figure 4.2.

Note that the robustness of the ECDH is in a hacker would have difficulties in obtaining either Ra or Rb despite the knowledge of RaP , RbP and P. However, it is susceptible to a Man-in-the-Middle attack, because these key messages can be intercepted by snooping on the channel during the authentication phase.

4.3 AVISPA Simulation Platform and Key Performance Evaluation Indicators

The Automated Validation of Internet Security Protocols and Applications (AVISPA) which was solely developed as a platform for the validation of security-sensitive protocols, will be relied upon in this work [41]. The objective is to formalize protocols by automatically validating them and detecting errors.

The validation is performed with the message exchange writing in the High-Level Protocol Language (HLPSL), which is organized in a sender/receiver style [41]. It supports asymmetric and symmetric encryption, cryptographic hash functions, non-atomic keys, and exponentiation. The code is divided into roles performed by the agents (or entities) involved in the authentication procedure.

AVISPA has four back-ends and two are used in the validation of the protocols proposed in

this work, the On-the-fly-Model-Checker (OFMC) and the Constraint-Logic- based Attack Searcher (CL-AtSe). The back-ends return “SAFE” if the verification judges the protocol message exchange safe and “UNSAFE” if any security properties were violated, and the protocol is vulnerable to attacks.

The OFMC back-end generates a binary tree with the decisions that can be executed by the protocol and return the following results, as described in [42]: ParseTime, the time took to analyze the system; SearchTime, the time took for the system to search for attacks; VisitedNodes, the number of nodes visited in the verification; Depth, the depth reached in the visit.

In the CL-AtSe back-end, each step is modeled by constraints on the adversary‘s knowledge, and the analysis is designed for a bounded number of protocol steps (loops). It translates the HLPSL of the protocol into constraints that can be used to find attacks [42]. It returns the following results, as described in [42]: Analyzed, number of loops analyzed; Reachable, number of steps reached by the analysis; Translation, the time took to translate the HLPSL code; Computation, time took in the analysis of the protocol.

Given that the goal of any security-related protocols to ensure complete security, we thus summarily list down some of the primitives that will be used in the evaluation of group protocols to be reviewed. These include:

- Mutual Authentication
- Forward/Backward Secrecy
- Confidentiality
- Non-Repudiation
- Anonymity
- Non-Traceability
- Device loss due to theft
- Impersonation Attack
- Replay Attack
- Denial of Service (DoS)
- Man-in-the-Middle Attack
- Computational Cost
- Communication Cost
- Energy Cost

4.4 An E-health D2D Communication Based Group Authentication Protocol

In this section, we analyze a cryptography-based mutual authentication and key agreement protocol that whose candidacy for E-health is explored. In exploring the protocol, we focus on its resilience as well as its abilities to provide security in terms of primitives outlined earlier such as resistance to attacks, confidentiality, and anonymity. We will also make a comparative performance analysis of this protocol in terms of computational complexity as well as communications overheads. The tendency to move towards energy efficiency networking prompts us to explore its efficiency in this regard as well.

By definition, E-health is an umbrella term for services and applications that aim to provide health services using the IoT as the main platform. For connectivity's sake, the IoT is accessible via the GSM cellular network. Typical E_health services and applications will include remote monitoring of patients, via networked dedicated sensors. In some cases, several of these sensors would be embedded within the body and interconnected via a Body Area Network (BAN). With the "advent of working from home" gaining momentum, E-health now extends to remote diagnosis as well as the provisioning of health services to patients. D2D communication standards and protocols will facilitate medical devices' interconnectivity in the realization of the various innovative E-health-related services and applications. Several otherwise catastrophic resulting medical conditions such as heart seizures (attacks) and high blood pressures can be closely monitored with complete privacy. In this regard, the collaborative work on D2D Communication standards is ongoing under the umbrella of 3GPP by way of technical specifications and reports [43].

In parallel, lots of research is being carried out to further enhance both privacy as well as general security for E-health-based services and protocols. For instance, cloud server-based E-health services, applications, and related authentication protocols are explored in [44], [45]. A symmetric cryptology-based authentication protocol is discussed in [40], whereas the studies in [38] mitigate asymmetric cryptology approaches. Both studies seem to follow the same procedural steps, in ensuring that the authentication process and operation are similar and thus the following phases are defined: Initialization, Registration, and Authentication. Physical security comparisons however reveal that the symmetric cryptology-based protocol is vulnerable as the patient's device is not completely secured from theft. At the semantic level, the same protocol displays compromised confidentiality.

Similarly in [46], asymmetric and symmetric cryptography-based protocols accomplish four

phases of operation namely; hospital uploading (HUP), patient uploading (PUP), treatment and prescription (TP), and routing checkup (CP) are proposed. The authors assume the existence of a cloud server that will act as a storage of all retrievable medical-related data mostly collected from sensors. Once again some issues were identified about the security capabilities of both protocols. E.g. the protocol in [46], has physical security issues as the preservation of system anonymity cannot be guaranteed once the patient's device is lost due to theft, neither is the same protocol immune Denial of Service (DoS) attacks.

Table 4.1: Comparing of a few selected protocols [46]

schem	D2D communication	*E – Health*?	cyphering type?	cloud based?
[49]	No	Yes	Asymmetric	Yes
[40]	No	Yes	Asymmetric	Yes
[48]	Yes	No	Asymmetric	No
[47]	Yes	Yes	Asymmetric	No
[46]	Yes	Yes	Symmetric	Yes

In [47], [50] a hash Message Authentication Code (HMAC) based authentication protocols for D2D communication are presented. The same study goes on to further extend the studies to developing an Identity-Based Signatures (IBS) version protocol. In [48] two group authentication protocols; one formulated around Identity-Based Encryption (IBE) and the other based on DHKE are investigated.

In [51], the authors develop an optimized direct discovery model for the establishment of D2D communication links. In their formulation, they do make its functionalities as similar as possible to the ProSec protocol standard developed by 3GPP. The authors in [47] presented an Elliptic Curve Discrete Logarithm Problem (ECDLP) based m-health authentication scheme for D2D communication. It is a certificate less encryption scheme (CLGSC) that protects ongoing sessions from eavesdropping. Table 4.1provides a summary comparison of the various authentication protocols.

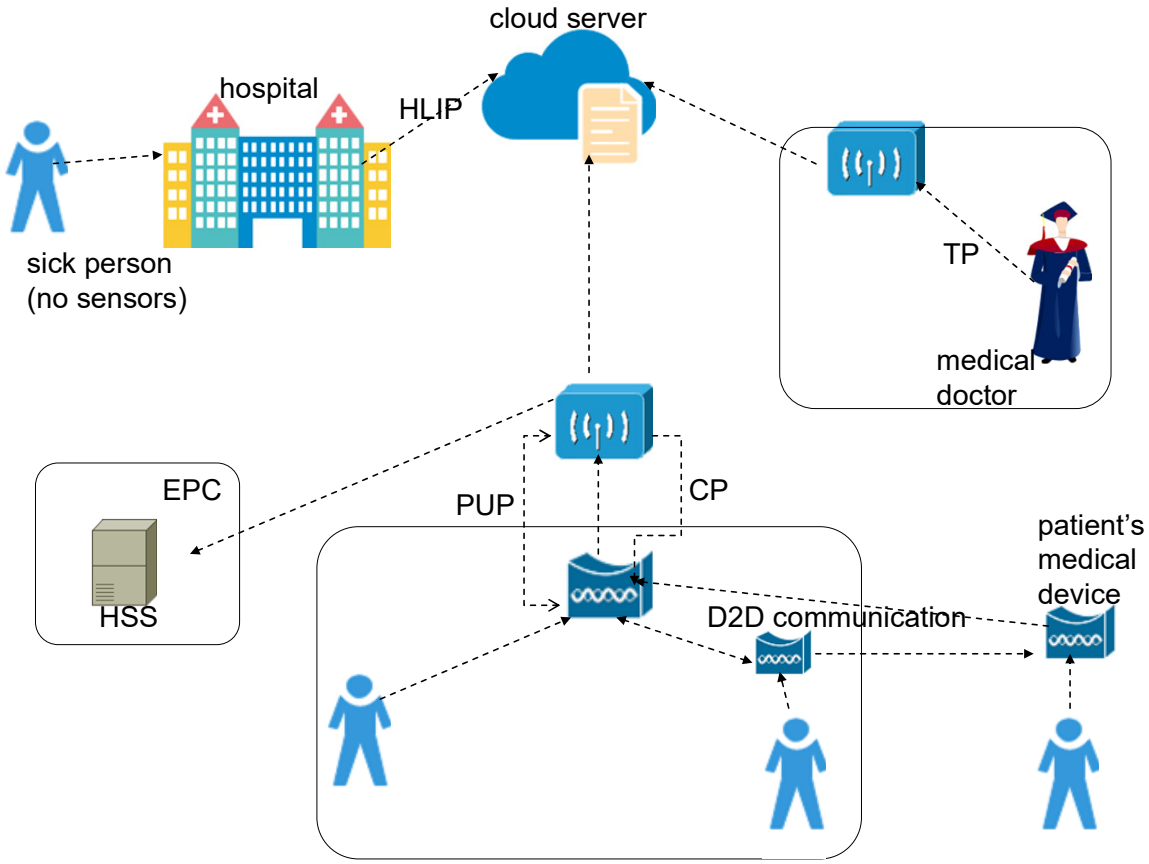


Figure 4.3: Health system infrastructure

We now proceed to describe, discuss as well as analyze the Group Authentication in a D2D Communication Based E-Health Protocol which among other things, enables the exchange of substantial volumes amounts of data. It was proposed in [46] and it is symmetric cryptography based. It is illustrated in Figure 4.3.

The system comprises patients, a hospital, a cloud server as well as communication infrastructure. The communication infrastructure is enhanced with 3GPP access technology. Typical units which form part of the communication infrastructure include a Home Subscriber Server (HSS), Evolved Node B (eNB), and a 3GPP Evolved Packet Core (EPC).coverage area (patients located outside the coverage area can access the 3GPP network relaying their data through devices located inside the coverage area), and the 3GPP domain, where the doctor is located. New patients visit the nearest hospital for registration purposes. Their furnished information will be used for authentication purposes in the future, e.g. authentication with the cloud server is mandatory before a patient's data can be uploaded or retrieved.

Table 4.2: Notations used in the protocol

symbol	inf *erence*
x, y	Key entities: patient (P), health center (H), doctor (D), cloud server (C).
ID_x / TID_x	the real identity of entity x/ Temporary identity of entity x.
k	random numbers are generated in the registration phase.
R_k	k random number generated.
MAC_{xy}	Message Authentication Code generated from entity x to entity y.
R_x	random number generated by entity x.
R_{cy}	random number generated by the cloud and sent to entity y.
T_x	timestamp generated by entity x.
K_{xy}y	session key generated by entities x and y.
C_{xy}y	validator of the session key generated by x and y.
E_{Kxy} / D_{Kxy}	Encryption/Decryption operation that used the session key generated by x and y
$IMSI_x$	International Mobile Subscriber Identity of device x
h_1	temporary identity generation hash function.
h_2	MAC generation hash function.
h_3	session key generation hash function.
h_4	session key verifier hash function.
⟶	secured channel
⇢	insecure channel

Likewise, at the devices level, each device performs a mandatory mutual authentication with the cloud server before any data exchange sessions can be sanctioned. Not all devices are within the 3GPP coverage infrastructure. Only those within its coverage may use it to access the cloud server. Otherwise, those which are not within coverage ranges rely on D2D communication to perform mutual authentication, before dispatching any patient-related reports. Relays will also assist other D2D devices to reach the cloud servers. Devices that connect directly to the 3GPP infrastructure may also utilize D2D communication for data exchanges with the cloud and other key parties. The medical personnel, namely medical doctors are also expected to mutually authenticate before gaining access to patients' records.

We next summarize the mandatory details of the multi-phase mutual authentication between the various entities (including patients and devices) and the cloud server as follows: used.

4.4.1 Device Discovery Scheme

For the invoking of any service, which might include several devices, each of the group members must perform neighbor device discovery to identify collaborating devices within the vicinity. Each device does that via the HSS. The HSS will in turn verify whether its International Mobile Subscriber Identity (*IMSI*) matches the device records in its database and whether the device is indeed is authorized (privileged) for the intended service, plus it is D2D communication compliant. Upon successful verification, the authorization will be relayed to eNB and at the same time tagged with a timer.

Next, all verified devices belonging to a group can now mutually identify each other as belonging to that group by possibly using WLAN direct radio signals is sharing their attributes.

4.4.2 Registration Phase

Key mutual authentication-related primitives are exchanged during this phase, The IMSI of each device must be registered in the HSS and normally this is done by the vendor. All key personnel and patients are registered to the cloud server via some identified secured channel. Each device is assigned a temporary identity $TID_y = h_1(ID_y \| R_k)$ where R_k is an arbitrary chosen random number will remain mapped to their real identities ID_y; h_1 is a hash function for the TID_y generation.

4.4.3 Hospital Uploading Phase (HUP)

This phase is exclusive to registered entities. An insecure channel is considered for this phase. The aim is mutual authentication among entities for the secure transmission of the patient's collected data, from the hospital to the cloud server. The complete procedure is shown in Figure 4.4. The phase starts when the user goes to the hospital for a health inspection and receives a login and a password to access the patient's system on his/her mobile device. Patients can access his/her health information whenever wanted by inserting the login/password pair on their device.

The overall authentication at this phase is sequentially carried out as follows:

Using a randomly generated integer R_h and its real identity ID_h the hospital computes its own message authentication code (MAC_h) as follows:

$$MAC_{hs} \rightarrow h_2(|ID_h|)(R_h) \tag{4.16}$$

Where, h_2 is a has function for generating the MAC. Later, the key primitives are relayed to the cloud server in the form of a time-stamped (T_h), message (m_1);

$$m_1 \rightarrow (TID_h, R_h, MAC_{hs}) \tag{4.17}$$

Upon receiving m_1 together with T_h from the Hospital the cloud server performs the necessary validations by initially computing:

$$MAC_{hs}' \rightarrow h_2(ID_h // R_h) \tag{4.18}$$

Note that the validation of (4) above is done using the real and temporary identities furnished by the hospital at the registration phase. It, therefore, suffices to compare its own computed MAC and that contained in m_1.

$$MAC_{hs}' \rightarrow MAC_{hs} \tag{4.19}$$

Upon successful authentication, it goes on to select a random number R_{sh} before computing.

$$MAC_{sh} = h_2(ID_h // R_{sh}) \tag{4.20}$$

This will now be sent as a time-stamped (T_s) confirmation message (m_2) back to the hospital.

$$\text{m2} \rightarrow (MAC_{sh}, R_{sh}) \tag{4.21}$$

Upon receiving the time-stamped message m_2 from the server, likewise, the hospital performs all the necessary validations as follows:

First, it checks that the validity of the received timestamp. This is followed by re-computing of the following.

$$MAC_{sh}' \rightarrow h_2(ID_h // R_h) \tag{4.22}$$

After which it verifies that the two MACs match.

$$MAC_{sh}' \rightarrow MAC_{sh} \tag{4.23}$$

Subject to the validity of (4.22), the next step would be to generate a common session key.

$$K_{hs} = h_3(ID_h // R_{h // R_{sh}}) \tag{4.24}$$

Where h_3 once again is a MAC generation function; R_{sh} is a randomly generated number by the cloud and sent to the hospital. A session key validator is also computed, with the help of a session key generator hash function h_4 as follows:

$$C_{hs} = h_4(K_{hs}) \tag{4.25}$$

The session key is used to cipher the patient's records before uploading to the server in the form of a message $m_3 = M_{rp}$;

$$M_{rp} \rightarrow E_{K_{hs}}(patient record, TID_h, C_{hs}) \tag{4.26}$$

The message m_3 is then time-stamped from the hospital side before dispatching it to the cloud server.

Upon receipt of m_3 and T_h the cloud server computes the session key

$$K_{hs} \rightarrow h_3(ID_h // R_h // R_{sh}) \tag{4.27}$$

and deciphers the patient's report.

$$(patient's\ record, TID_R, C_{hs}) \rightarrow D_{K_{hs}} \tag{4.28}$$

Ultimately it computes, $C_{hs} = h_4(K_{hs})$ and validates: $C_{hs}' = C_{hs}$

Only then will the records be admitted to the Cloud server's database.

HUP	
hospital	*cloud server*
step I *select* TID_H *generate* R_H $MAC_{HC} \rightarrow h_2(\lvert D_H \rvert)(R_H)$ $message1 \rightarrow (TID_H, R_H, MAC_{HC})$ *selects* T_H →	
	step II *check* T_H $MAC_{HC}' \rightarrow' h_2(ID_H \,//\, R_H)$ $MAC_{HC}' \rightarrow MAC_{HC}$ *select* R_{CH} $MAC_{CH} = h_2(ID_H \,//\, R_{CH})$ $message\ 2 \rightarrow (MAC_{CH}, R_{CH})$ *select* T_C
←	
step III *check* T_C $MAC_{HC}' \rightarrow' h_2(ID_H \,//\, R_H)$ $MAC_{CH}' \rightarrow MAC_{CH}$ $K_{HC} = h_3(ID_H \,//\, R_H \,//\, R_{CH})$ $C_{HC} = h_4(K_{HC})$ $M_{RP} \rightarrow E_{KHC}(patient\ record, TID_H, C_{HC})$ $message\ 3 \rightarrow M_{RPC}$ *select new* T_H	
→	
	step IV *check* T_H $K_{HC} \rightarrow h_3(ID_H \,//\, R_H \,//\, R_{CH})$ $(patient's\ record, TID_R, C_{HC}) \rightarrow D_{KHC}$ $C_{HC}' = C_{HC}$ $store \rightarrow patient's\ record$

Figure 4.4: Hospital Uploading Phase Message Exchanges

4.4.4 Patient Uploading Phase (PUP)

This involves uploading all acquired patient's health-related data via sensors to be uploaded to the cloud server. This can be done over a generally insecure channel since most patients are scattered around the countryside and with an insufficient network (3GGPP)/ coverage. This data will thus be encrypted for confidentiality's sake. The patient's main device will prompt all sensors within the vicinity (around his/her body) to release the data to it. Authentication is necessary before the data collection by the main patient's device initiates. It is important to note that all associated devices must be initially successfully authenticated with the available 3GPP network.

If they are to utilize the D2D communication mode, then a random number R_p is generated by the patient's device. It uses this same number to compute a hash of its $IMSI$.

$$Auth_p = h_1(IMSI_p \| R_p \tag{4.29}$$

Ultimately the hash is sent to the HSS for use in the authentication verification according to:

$$Auth'_p = hi(IMSI_p \| R_p) = Auth_p \tag{4.30}$$

Once authenticated by the HSS the patient's device can perform discovery with peer proximity devices using their temporary devices as well. It is also possible for a device that is outside a 3GPP coverage area to rely on close-by devices to access the 3GPP network coverage range and ultimately authenticate with the cloud server.

Figures 4.5 and 4.6 further detail the procedural steps.

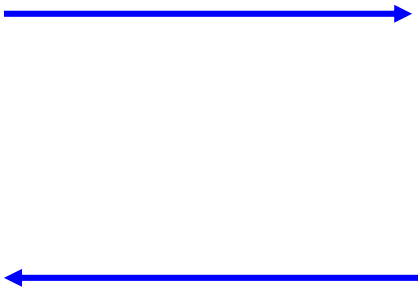
<table>
<tr><td colspan="2">PUP</td></tr>
<tr><td>$patient\_device$</td><td>$cloud\_server$</td></tr>
<tr><td>$step\_I$
$slecet\_TID_p$
$generate\_R_p$
$MAC_2(ID_p // R_P // f_{Pp} // ECG_p)$
$message\_1 = (TID_p, R_p, MAC_{pC})$</td><td></td></tr>
<tr><td>  </td><td>$step\_II$
$checks\_T_p$
$MAC_{pC}' = h_2(ID_p // R_{Cp} // f_{Pp} // ECG_p)$
$MAC_{pC}' = MAC_{pC}$
$selects\_RC_{Cp}$
$MAC_{Cp} = h_2(ID_p // R_{Cp} // f_{Pp} // ECG_p)$
$messag\_2 = (MAC_{Cp}, R_{Cp})$
$select\_T_c$</td></tr>
<tr><td>$step\_III$
$checks\_T_C$
$MAC_{Cp}' = h_2(ID_p // R_{Cp} // f_{Pp} // ECG_p)$
$MAC_{Cp}' = MAC_{Cp}$
$K_{pC} = h_3(ID_p // R_p // R_{Cp})$
$C_{pC} = h_4(K_{pC})$
$M_{MS} = E_{KPC}(sensor\ measurements, fingerpr\,\mathrm{int}, ECG, TID_p, C_{pC})$
$message\_3 = M_{MS}$
$select\ \ new\_T_p$</td><td></td></tr>
<tr><td>  </td><td>$step\_IV$
$check\_T_p$
$K_{pC} = h_3(ID_p // R_p // R_{Cp})$
$(sensors\_measures, fingerpr\,\mathrm{int}, ECG,$
$TID_p, C_{pC}) = D_{KPC}(M_{MS})$
$C_{pC}' = C_{pC}$
$stores\ sensor\ \ measurements$</td></tr>
</table>

Figure 4.5: Message exchange in PUP for direct access to 3GPP infrastructure

Figure 4.6: Message exchange in PUP

.Note that in this case, the message exchange in PUP illusttated in Figure 4.6 is when D2D communications is adopted to reach the 3GPP infra-structure and the cloud server.

4.4.5 Treatment Phase (TP)

Once again, the two parties involved; the medical specialist and cloud server must mutually authenticate. Ultimately a session key will be generated and used to cipher all patient's reports, body sensor measurements as well as the overall diagnosis. The procedures taken are as follows.

The medical specialist must initiate the authentication with the server by generating a random number R_d as well as providing a temporary ID (TID_d). The two will be used to generate:

$$MAC_{ds} = h_2(ID_d // R_d) \tag{4.31}$$

Which is now time-stamped (T_d) and dispatched to the server in the form of a message m_1.

$$m_1 = (TID_d, R_d, MAC_{ds}) \tag{4.32}$$

Upon receiving both m_1 and T_d the server validates them before further computing

$$MAC_{ds}' = h_2(ID_d // R_d) \tag{4.33}$$

and ultimately verifying;

$$MAC_{ds}' = MAC_{ds} \tag{4.34}$$

Should the verification succeed, the server once again opts a random integer R_{sd} and uses it to compute the MAC and session key:

$$MAC_{sd} = h_2(ID_d // R_{sd}) \tag{4.35}$$

$$K_{ds} = h_3(ID_d // R_d // R_{sd}) \tag{4.36}$$

$$C_{ds} = h_4(K_{ds}) \tag{4.37}$$

The server uses the session keys to cipher the patients records it retrieves from the database:

$$M_{RpMS} = E_{KHC}(p\_RR, sensor, TID_p C_{ds}) \tag{4.38}$$

medical _ doctor	*cloud _ server*
step _ I *select _ TID_p* *generate _ R_D* *$MAC_{DC} = h_2(ID_D // R_D)$* *message _ 1 = (TID_D, R_D, MAC_{DC})* *select _ T_D*	
	*step _ II* *check _ T_D* *$MAC_{DC}' = h_2(ID_D // R_D)$* *$MAC_{DC}' = MAC_{pDC}$* *select _ R_{CD}* *$MAC_{CD} = h_2(ID_D // R_{CD})$* *$K_{DC} = h_3(ID_D // R_D // R_{CD})$* *$C_{DC} = h_4(K_{DC})$* *$M_{RpMS} = E_{KDC}(patient'sreport, sensor\_measurements, T$* *messsage _ 2 = $(MAC_{CD}, R_{CD}, M_{RpMS})$* *select _ T_C*
 *step _ III* *check _ T_C* *$MAC_{CD}' = h_2(ID_D // R_{CD})$* *$MAC_{CD}' = MAC_{CD}$* *$K_{DC} = h_3(ID_D // R_D // R_{CD})$* *$(patient's\_reports, sensor\_measurements, TID_D, C_{DC}) = D_{KI}$* *$C_{DC} = h_4(K_{DC})$* *$C_{DC}' = C_{DC}$* *$M_{Digs=E_{KDC}}(medicaldoctor'sdignosis, TID_p, C_{DC})$* *message _ 3 = $(M_{diag.})$*  *select _ new _ T_D*	
	step _ IV *check _ T_D* *$K_{DC} = h_3(ID_D // RD_D // R_{CD})$* *$(medical\_doctor's\_diagnosis, TID_p, C_{pC}) = C_{DC} = D_{KDC}$* *$C_{DC}' = C_{DC}$* *stores _ medical _ doctor 's _ diagnosis*

Figure 4.7: Message exchange during the treatment phases

Finally, it timestamps (T_s) the records and sends them in the form of a message m_2 to the medical specialist.

$$m_2 = (MAC_{sd}, R_{sd}, M_{RpMS}) \tag{4.39}$$

Upon receiving m_2 and T_s, the medical specialist validates them and ultimately generates a session key:

$$K_{ds} = h_3(ID_d // R_d // R_{sd}) \tag{4.40}$$

He will then decipher the received patient records before sending a time-stamped confirmation message (m_3) back to the cloud server. The latter will have to positively validate m_3 otherwise this is a malicious activity of the side of the medical specialist (i.e., a hacker is attempting to infiltrate the system). Figure 4.7 details the entire process.

4.4.6 Routing Check-up Phase (CP)

This is a four-step phase detailed in Figure 4,8.

Step I. A patient computes time-stamped mandatory request message 1, using a randomly generated number before sending it to the cloud server. The communication can be executed as D2D provided the patient is under 3GPP network coverage.

Step 2. Likewise, the cloud server will verify the authenticity and validity of the received message 1, and if successful it generates a session key. The session key will be used for decryption purposes once the stored Medical Doctor's records on the patient are retrieved.

Step III. Upon successfully receiving *message 2, the* patient deciphers it using their own generated session key. The patient upon usefully decrypting the Medical Doctor's reports can proceed with medication.

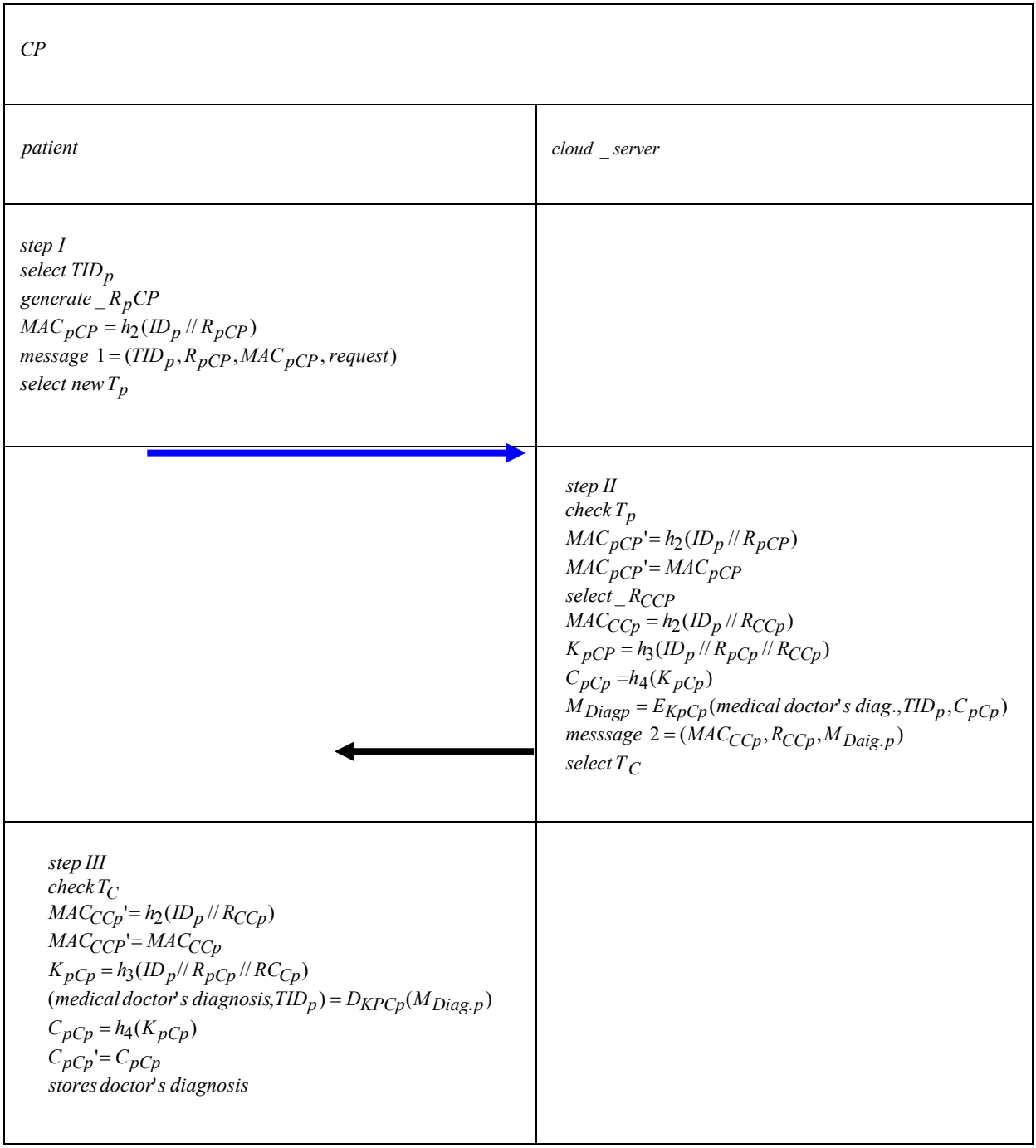


Figure 4.8: Message exchange in CP

4.4.7 Security and Performance Analysis

We summarily analyze the protocol in terms of security requirements as well as performance. The performance is restricted to computational simplicity, communications overhead as well as energy efficiency.

4.4.7.1 Mutual Authentication

With the protocol, any two communicating parties reciprocate each other in computing the MAC for mutual authentication purposes.

4.4.7.2 Forward/Backward Secrecy

The protocol relies on the generation of random values (RH, RCH, RP, RCP, RD, RCD, RPC, RCPC) in each initiated session hence the old system keys are not valid for future sessions. In that way, backward secrecy is guaranteed. Similarly, keys intended for future use are not valid for use in past authentication sessions henceforward secrecy is guaranteed.

4.4.7.3 Confidentiality

The protocol runs authentication scripts for every session. Specifically, each session's key generation is robustly authenticated.

4.4.7.5 Non-Repudiation

The use of temporary identities by all parties) and restricting the knowledge of real identities to the cloud server ensures non-repudiation.

4.4.7.6 Anonymity

Assigning and relying on temporary identities for authentication purposes ensured anonymity. The cloud server is secluded in the authentication process hence the reliance on insecure channels for the initial authentication does not compromise its identity.

4.4.7.7 Non-Traceability

Periodically changing temporary identities or assigning a set to each entity ensures non-traceability. This is further enhanced with the use of randomly generated numbers for each authentication procedure (session).

4.4.7.8 Session Key Security

Session keys are localized and not exchanged. In that way, they cannot be intercepted along compromised channels in case they were exchanged via such channels.

4.4.7.9 Impersonation Attack

The lack of knowledge of the real identities of both the cloud server and the rest of the entities means intruders or attackers have no chance of succeeding impersonating them. Furthermore, a valid MAC is a function of the associated entity's real identity.

4.4.7.10 Replay Attack

Random values are constantly generated for computing new session keys and other authentication-related primitives hence this secludes the possibility of an attacker succeeding to forge messages utilizing old values.

4.4.7.11 DoS

Usage of time stamping throughout secludes the possibility of DoS attacks.

4.4.7.12 M-in M Attack

Authentication is accomplished using exchangeable values and localized (un-exchangeable) values. In that way Main-in-the –Middle attacks are impossible to execute.

- Performance Analysis - The performance analysis briefly compares the computational complexities, communication overheads as well as energy efficiencies of selected protocols for the schemes presented in. [40], [49] and [46].
- Computational Cost - The execution times of the operations in the various schemes are presented in Table 4.4. 2GB; operational system: Windows 7 Professional.

Table 4.3: Execution time of each operation considered

symbol	*defination*	cos t(sec s)
Ts	signature verification	0.3317s
T_p	pairing	0,0621s
T_E	symmetric cyphering/deciphering	0.0087s
T_H TH	one way Hash function	0.0005s

Compared in the table above are computational costs among protocols proposed in [40], [49], and [46].

Table 4.4: Computational Cost of the Protocols

	[49]	[40]	[46]
HUP	$nT_S+3nT_P+2nT_E+7nT_H$	$nT_S+3nT_E+11nT_H$	$2nT_E+8nT_H$
PUP	$nT_S+3nT_P+2nT_E+9nT_H$	$2nT_S+2nT_E+10nT_H$	$4nT_E+9nT_H$
TP	$2nT_S+3nT_P+2nT_E+8nT_H$	$2nT_S+2nT_E+9nT_H$	$4nT_E+8nT_H$
CP	$nT_S+2nT_P+2nT_E+8nT_H$	$nT_S+2nT_E+5nT_H$	$2nT_E+8nT_H$
TOTAL(s)	$5nT_S+11nT_P+8nT_E+32nT_H = 2.43n$	$4nT_S+9nT_E+35nT_H = 1.42n$	$12nT_E+33nT_H = 0.21n$

Figure 4.9 plots the relative computational costs in which we see that the scheme in [56] required the lowest computational cost. This is attributed to the fact that symmetric cyphering/deciphering for the authentication. NB: Symmetric ciphering/deciphering has the advantage of low communication cost and at the same time it can perform the necessary operations in an energy-efficient manner as well.

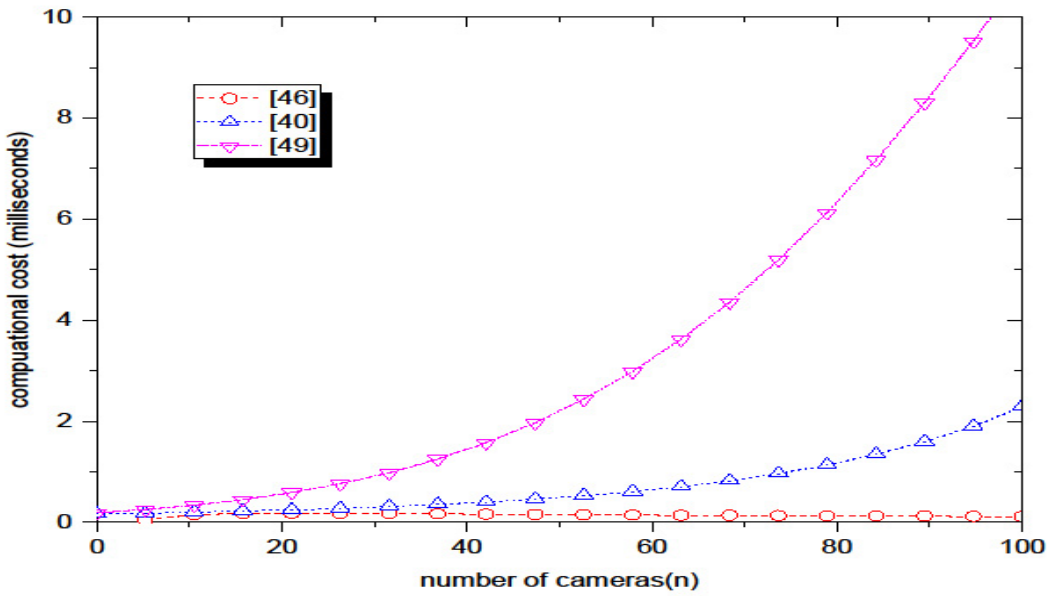


Figure 4.9: Computational cost comparison

4.4.8.2 Communication Cost

D2D communication cost metrics are analyzed the volume of exchanged over an insecure link Associated parameters corresponding costs are shown in the table below.

Table 4.5: Parameters and costs in bits

parameter	cos *t*
Random Number/Identity/Timestamp	48 bits
Bilinear Pairing/Hash	148bits
Symmetric Key	128 bits
Signature (symmetric algorithm)	1024 bits

Once again only the schemes (protocols) discussed in [49], [40] and [46] are compared.

Table 4.6: Comparison of communication costs in bits

	[49]	[40]	[46]
HUP	699n	480n	699n
PUP	1500n	1650n	699n + 699m + 20(m-1)
TP	2002n	1700n	850n
CP	1525n	1097n	699n

The protocol proposed in [46] generates relatively low communication overheads hence it is best suited for adaptation to D2D communication.

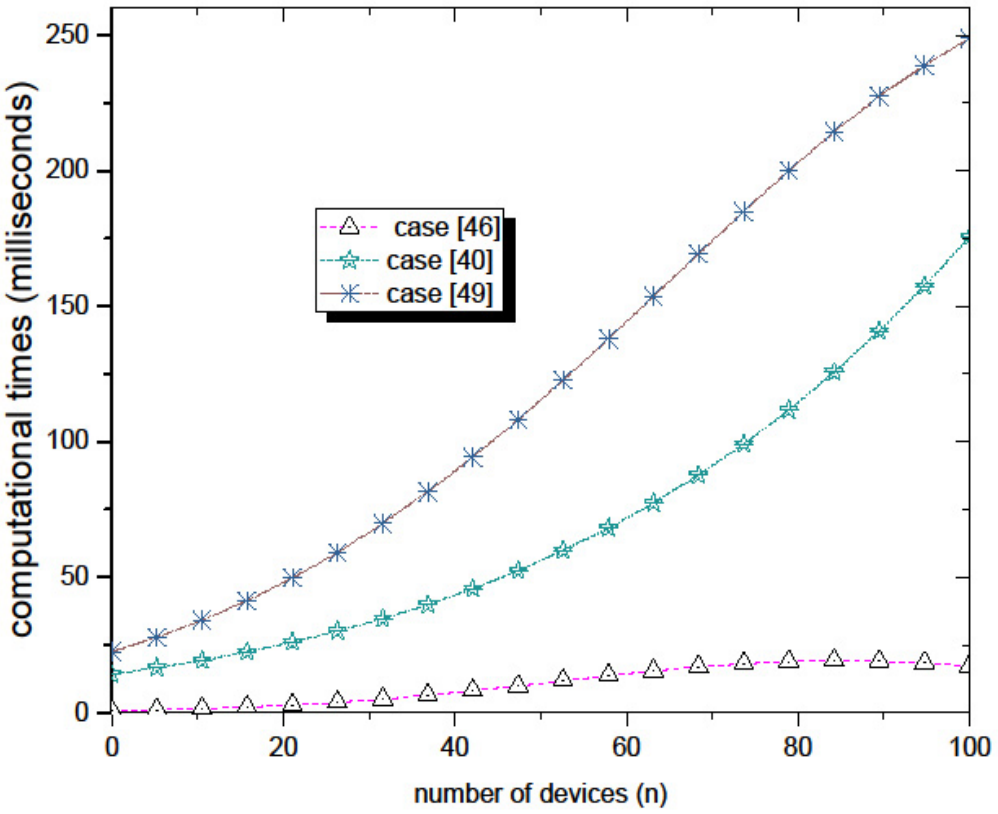


Figure 4.10: Communication cost comparison

The schemes in [40] and [49] incur higher communication costs since they involve the exchanging of quite a few costly signature parameters, for their mutual authentication, which is reduced in the traditional 3GPP network.

4.4.8.3 Energy Cost

Most of the energy consumption is incurred by the CPUs of the respective devices.

Table 4.7: Energy cost of protocols

	[49]	[40]	[56]
aggregate	(4nTS + 10nTP + 7nTE + = 26.43n mJ	(4nTS + 9nTE + 35nTH = 15.45n mJ	(13nTE + 22nTH) = 1.32n mJ

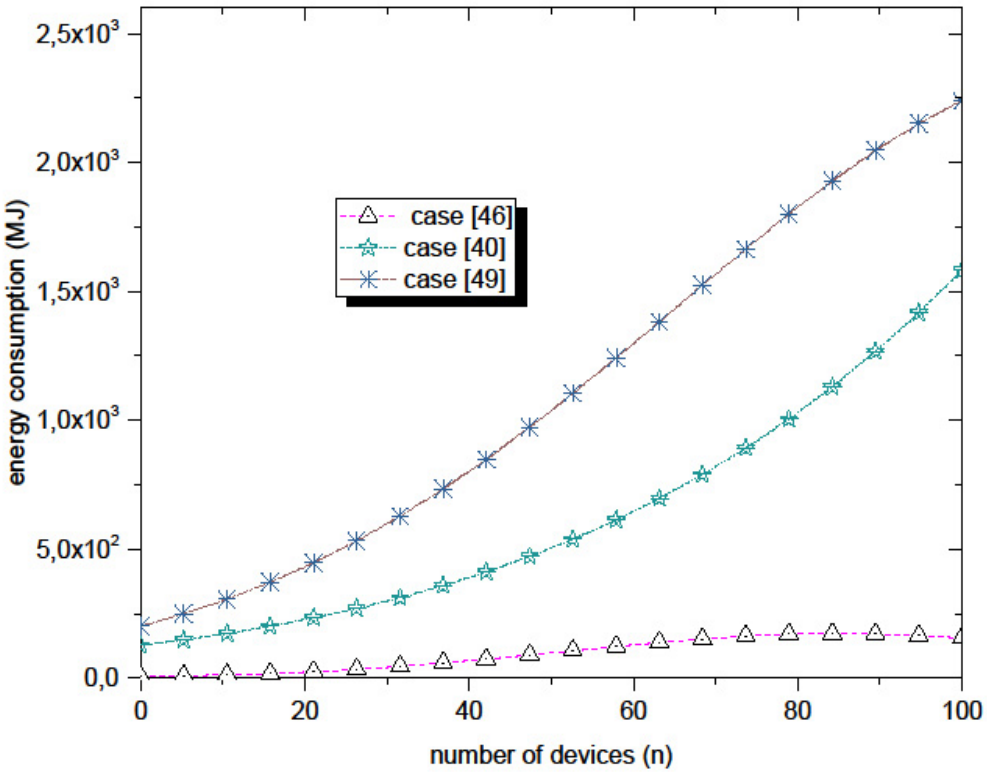


Figure 4.11: Energy cost comparison

The energy cost for the various protocols is calculated as set out in Table 4.8. By comparison, the plot of Fig 4.11 shows that the scheme in [46] is more energy efficient.

4.5 Lightweight Cryptography Based Authentication

As is known D2D communication is quite advantageous in mobile communication networks as it enhances the energy-efficient operation of the overall network, by way of transmitting data between devices rather than via the base station. Furthermore, it helps reduce interference problems since the separation distance between adjacent devices is often shorter than that between a BS and a device. It is generally noted that the magnitude of radio frequency interference is proportional to distance, hence better data multiplexing efficiency can be achieved with D2D communication.

A lot of would-be D2D communications-based applications and services will use the IoT as the core communication infrastructure. In itself, the IoT has a heavy reliance on wireless network infrastructures, let alone the peripheral sections. So the IoT devices are always resources constrained and thus it makes it difficult to implement security measures in such scenarios. It is thus important to secure D2D communication in IoT by provisioning proper authentication in peer devices. Lightweight cryptography is known to be an appropriate solution

for implementing security resource-constrained environments and devices. E.g. Elliptic curve cryptography (ECDH), which is an example of lightweight asymmetric-key algorithms uses relatively smaller encryption keys in comparison to their public-key encryption algorithm RSA counterparts [52]. In this section, we explore a lightweight cryptography ECC and the AEAD cipher based authentication protocol for D2D communication [53].

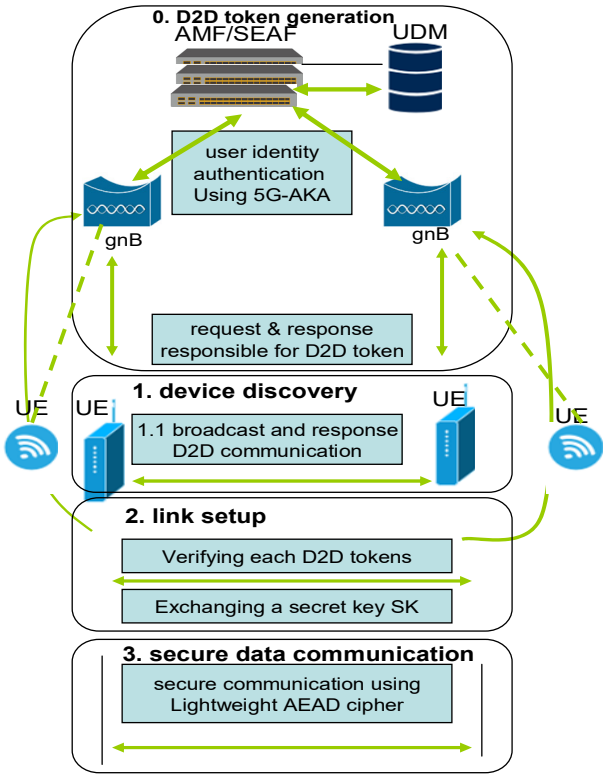


Figure 4.12: A 5G/IoT Secured D2D communication system model

Such a model is provided in Figure 4.51. Key elements of the system model are general Node-B (gNB) which is responsible for interconnecting the various network elements, user equipment (UE) which is a mobile device that carries out the communications., access, and mobility management function (AMF) which manages the mobile entities, /security anchor function (SEAF) which plays the role of a middle entity of authentication between UE and a 5G network and is co-located with AMF. Finally, we have a user data management (UDM) whose sole purpose is to store information about mobile entities in a 5G network. It is assumed that a public key is shared among gNBs in the network each uses its own generated private key to generate D2D tokens ($D2DTK_{gNB_X}$) via an elliptic curve digital signature algorithm (ECDSA). Verification of the various UE elements facilitated by the existing 5G authentication and key agreement (AKA) framework. Once authenticated, the UE is provided with a D2D token for communications purposes. Post D2D token generation, the communication it-

self will follow a three-phase approach namely; Device discovery, link setup, and secure data transmission. Each phase incorporates security features such as anonymity, authentication, and confidentiality/integrity.

Summarily the three phases are described as follows:

- Device discovery Phase: A device in the networks broadcast a discovery message and nearby devices will respond to their respective identities, SUCI, and D2D token in encrypted form.
- Link setup phase: This prepared devices for peer-to-peer connection. Each device will dispatch a verification request to the nearest Bs (gNB) together with a SUCI and D2D token of the intended target UE in the discovery phase. The data is encrypted using a lightweight AEAD. Post verification (authentication) ECDH based ciphering is used to exchange secret keys for secure data transfer.
- The secure data transmission phase entails the exchange of encrypted data between peers. The sender device utilizes its D2D token identity and context sequence, thus both the integrity as well as the confidentiality of the data are guaranteed. Authentication is mandatory for every fresh transmission,

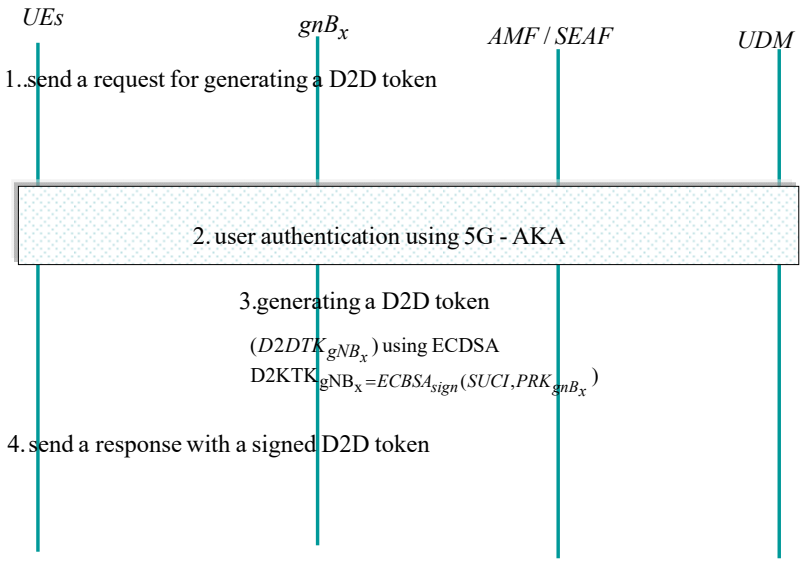


Figure 4.13: D2D token generation procedure

4.6 Security Analysis

In this section, we briefly analyze the security performance of the scheme. The performance

consideration will center on its efficiency in accomplishing authentication, confidentiality, integrity, anonymity, etc. despite operating in an adverse (resource constraint) environment. We summarize its performance as follows.

- Authentication: This does not rely on a single authentication phase, It utilizes the already existing 5G-AKA for primary authentication, (i.e authenticating a UE using before issuing a D2D token). The creation of a D2D link also requires verification. Using the gNB's public key and SUCI. Furthermore, the UE is continuously authenticated during data transmission phases.
- Data confidentiality and integrity: the UE's identity is encrypted throughout in order to ensure confidentiality and integrity. . This is accomplished by way of generating the D2D communication using SUCI. Furthermore, a D2D link, ciphering is is carried out using a lightweight AEAD cipher. AEAD cryptography is already well known for providing both integrity, authentication as well as data confidentiality.
- Anonymity: The scheme encrypts the UE's identity using SUCI hence the UE remains anonymous. Besides, the D2D token approach ensures that the gNB is unable to unmask the identity of the UE directly.
- Efficiency: The system utilizes and is based on lightweight ciphering and deciphering (1024-bit lightweight AEAD cipher and 256- bit ECC-based public key cryptosystem). Both are designed to operate efficiently in resource-constrained environments. And still, provide robust authentication as well as data confidentiality/integrity.
- Privacy sniffing. We recall that the D2D token is generated by way of utilizing a UE's SUCI and digital signature of gNB using ECDSA. Thus in that way, the token itself provides anonymity as a cryptographic identity. This makes it problematic for an attacker to decipher the real identity of a UE.
- Impersonation attack. An attacker is unable to impersonate another UE because each UE's token is issued as well as signed by a gNB only after an exhaustive authentication process.
- Free-riding attack and location spoofing. The UE's authentication is managed by a gNB. The D2D token is also authenticated during the link setup phase. This will ensure the total elimination of any malicious UEs.
- Eavesdropping. A lightweight AEAD cipher is used to secure each data transmission step.

4.6.1 Performance Evaluation

We briefly explore this scheme's relative performance and efficiency. Notably, for efficiency, we focus on relative implementation costs of lightweight AEAD ciphers, as well as energy consumption. The scheme is expected to provide adequate security and privacy and hence this necessitates consideration of its efficacy in terms of prime security primitives that we discussed previously, i.e. they include but are not limited to authentication, data confidentiality/integrity, anonymity e.tc. In detail, the applied cryptographic algorithms are the digital signature, the Diffie–Hellman key exchange algorithm, and the AEAD cipher. A few additional key parameters that we define to assist in the analysis include:

- t_{DS}- time required to process digital signature.
- t_{DS}-verification time for a single digital signature.
- t_{DH}- time lapse is required to process a key exchange.
- t_{AEAD}- the processing time for the AEAD cipher.
- l_{tr}- average transmission latency in D2D communication.

Thus the processing time of a D2D communication processing time is [53];

$$t_{D2D} = \sum l_{tr} + \sum t_{DS_{sign}} + \sum t_{DS_{ver}} + \sum t_{DH} + \sum t_{AEAD} \tag{4,50}$$

Table 4.8:Processing times of each key step

phase	transmsisi on delays	Processing time of Cryptograp hic Algorithm			
		ECDSA – sign	*ECDSA – verify*	*ECDH*	*AEAD*
0	$12I_{tr}$	$2\times t_{DSsign}$	$2\times t_{DSver}$	-	-
1	$(m+1)I_{tr}$	-	-	-	-
2	$6I_{tr}$	-	-	t_{DH}	-
3	$(n/1460)I_{tr}$	-	-	-	$2\times t_{AEAD}$
Total$(\sum l / or \sum t)$	$(19+m+n/1460)I_{tr}$	$2\times t_{DSsign}$	$2\times t_{DSver}$	t_{DH}	$2\times t_{AEAD}$

Table 4.8 summarizes the processing times of each key step of the following five AEAD ciphers:

- AES-GCM . This is based on a block cipher mode of operation that affords high speed of authenticated encryption and data integrity. The AES-GCM algorithm encrypts or decrypts with 128-bit, 192-bit, or 256- bit cipher key. CM mode provides both privacy (encryption) and integrity. To provide encryption, GCM maintains a counter; for each block of data, it sends the current value of the counter through the block cipher. Then, it takes the output of the block cipher, and exclusive or's that with the plaintext to form the ciphertext.
- ASCON: This is an AEAD version cipher algorithm based on duplex sponge modes. It uses a 128-bit key, a 128-bit IV, and produces a 128-bit authentication tag. The internal state is 320-bit long and is represented by five 64-bit registers, noted x0 to x4. Finally, the secret key is XORed to the 128 last bits of the state.
- SpoC (Sponge with masked Capacity.). This is a permutation-based mode of operation for authenticated encryption with associated data (henceforth "AEAD") functionality. The high-level design is inspired by the Beetle mode of operation. It offers a higher security guarantee with smaller states as compared to some of the previous AEAD designs based on the Sponge paradigm.
- Spook (Sponge-Based Leakage-Resistant Authenticated Encryption): This is an algorithm for authenticated encryption It is primarily designed to support low energy implementation, especially when protection against side-channel attacks is required. Spook is generally regarded as an efficient single-pass algorithm. \
- GIFT-COFB . This is primarily a block cipher-based AEAD design that uses GIFT-128 as the underlying block cipher. In its basic operation, it receives an (1) 128-bit encryption key K, (2) a 128-bit nonce N, (3) an associated data A of arbitrary length, (4) and a message M of arbitrary length as inputs, and returns a (5) ciphertext C of the same length as that of the message, and (6) a 128-bit tag T. Note that GIFT-COFB is an inverse-free authenticated encryption algorithm. Both encryption and decryption algorithms do not require any decryption call to the underlying block cipher. This significantly reduces the overall hardware footprint in combined encryption-decryption implementations.

Figure 4.14 plots the duration of the D2D communication. When transmitting a 10kB data message. Relatively the ASCON, Spook and GIFT-COFB are faster than AES-GCM.

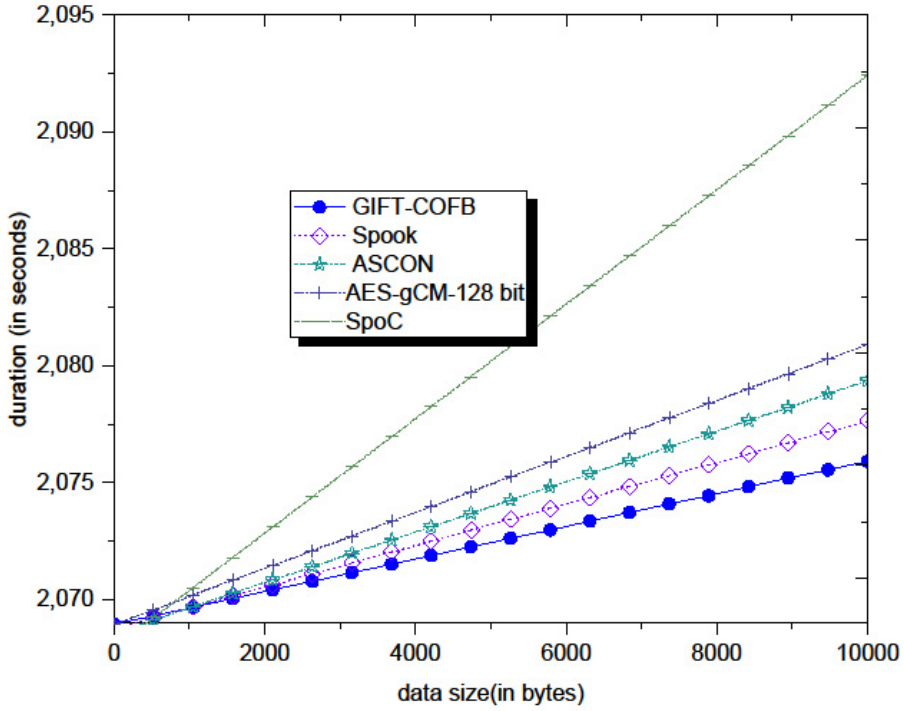


Figure 4.14: Processing time of the proposed D2D communication system

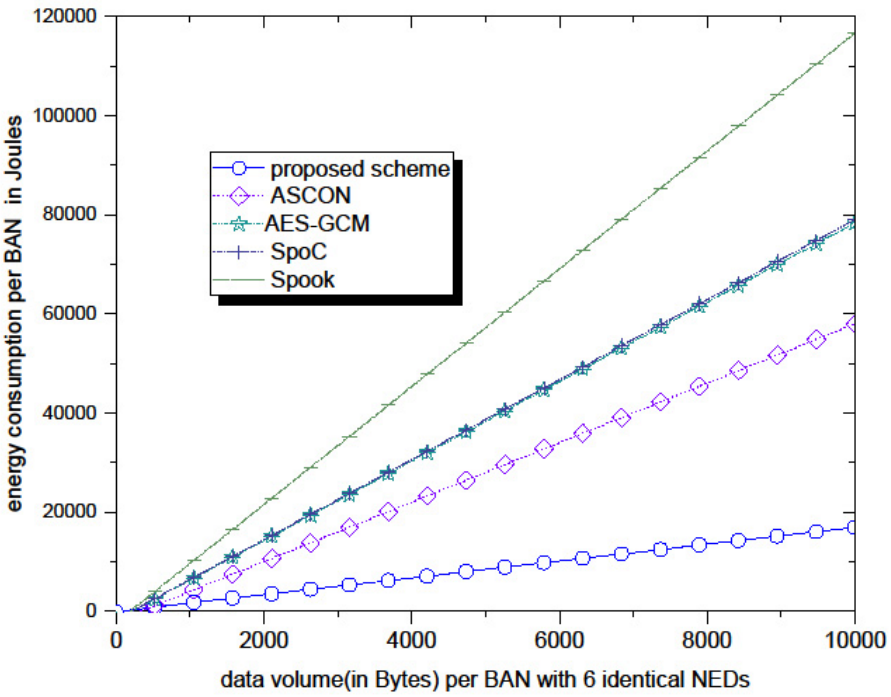


Figure 4.15: The energy consumption of AEAD ciphers

Figure 4. 15 plots the energy consumption comparisons. In the plot, GIFT-COFB and ASCON utilize relatively lesser energy in comparison with AES-GCM. However, both SpoC and Spook consume quite considerably. Taking into consideration that GIFT-COFB and ASCON display relatively superior performance when compared to AES -GCM they are therefore better suited for integrating with D2D communications technologies. This is because they both display superior turnaround times at the same time being energy efficient.

It can be concluded therefore that an ECC-based public key cryptosystem and a lightweight AEAD cipher authentication protocol are reasonably suited for resource constraint environments such as the IoT in 5G use cases corresponding to IoT.

4.7 Group AKA (Gr-AKA) protocol for D2D communication

The section commences by defining the system model as well as defining security assumptions. We then present the GAKA protocol. The system model is based on the generalized 3GPP MTC architecture as described in the previous section. We consider a conventional application scenario such as remote weather monitoring or crime surveillance. Initially, an MTC user registers with the local service provider for such D2D communications service. This is followed by the network identifying a group of MTCDs ($MTCD_{grp_i}$) in the targeted area and initializing them. The group then designates a group leader ($grp_{i-leader}$) who will, in turn, negotiate both authentication and key establishment with the HSS/MME on behalf of the group. During this phase, the MTCDs and the HSS authenticate via the MME. Session keys are established between MTCDs and HSS for the secure transmission of messages.

Session Key Compliance Stage: To ensure secured message exchanges between the MTCDs and the designated group leader, a session key is established among the group members. Because individual MTCDs may leave or exit the group; for each exit/or joining, key updating is necessary. A key generation center (KGC) communicates the updated information to all group member MTCDs.

MTCD join event: When a device joins a group, a new key is generated, so is the case when an existing member vacates the group.

MTCD exit event: A member can exit upon completing their task. In this case, it must be prevented from accessing the group's resources, otherwise, security is breached. Hence the necessity to update current keys.

4.7.1 System Assumptions

We make the following assumptions:

- Considered is an application or service in which several MTCDs together cooperate to form a group on one end and a single MTC user at the other end.
- The users are not necessary in their HOME location, and thus therefore prior registration (in case of roaming users) is necessary.
- The IoT service providers initially generate and agree on common system parameters as well as in-house and inter-operator agreements for D2D applications and services.

Asynchronous $(t; m; n)$ Group Authentication Scheme as proposed in [59] and further explored in [60] is utilized as the basis for carrying out group authentication.

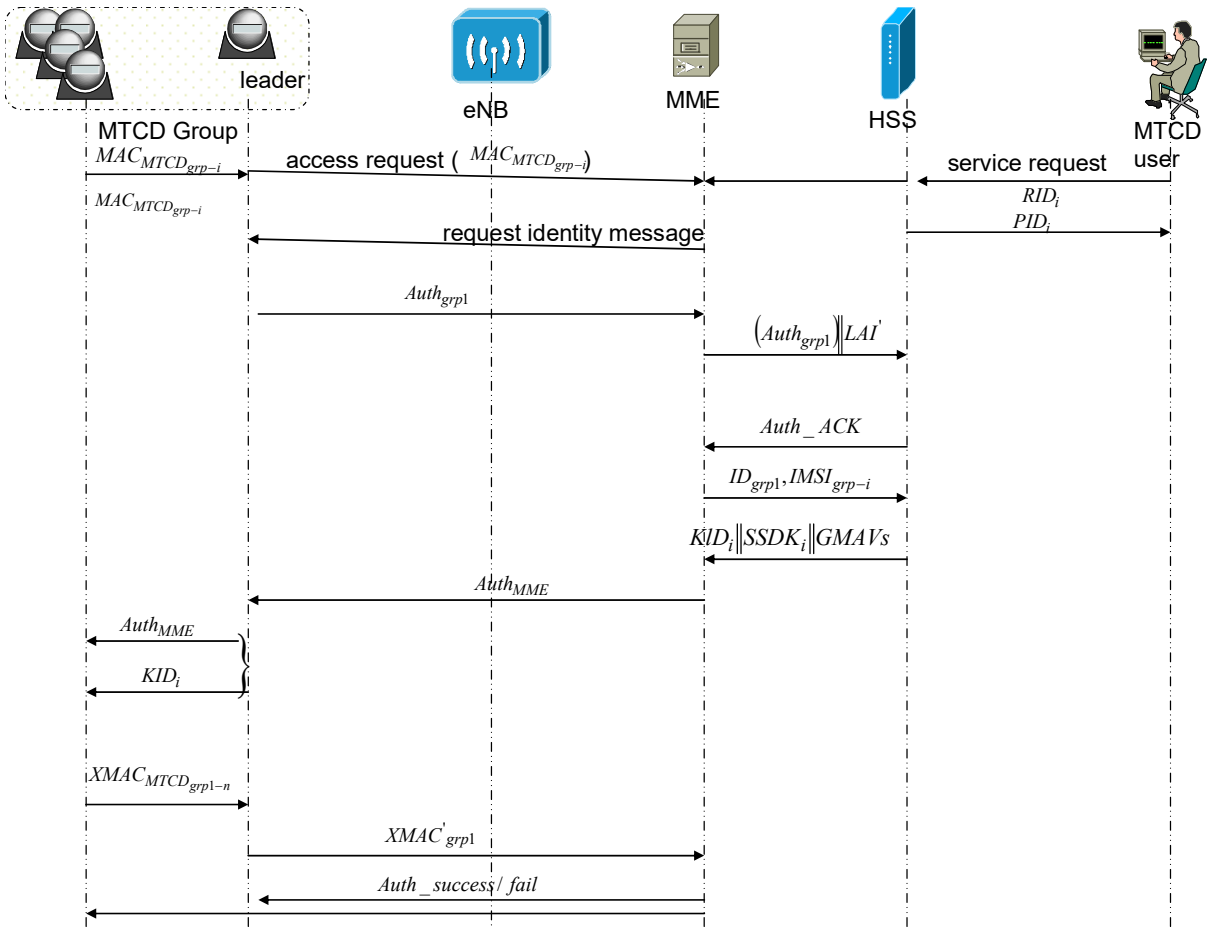


Figure 4.16: Sequence events for the proposed protocol

Asynchronous (t, m, n) group and authentication guarantee group authentication for m devices of a group with n members as well as being tolerant to t compromised tokens. In our proto-

col, it is considered that m has the same size n, that is, all the members in a group are authenticated. Thus, it authenticates all the devices in a group simultaneously. The various sequence of events is summarised in Figure 3. The detailed descriptions are narrated in the next section.

A: Session Request and Group Registration

A roaming MTC user registers for D2D services.

The MTC user (U_i) with a valid identifier (RID_i) performs user registration with the local HSS by furnishing his/her RID_i. If the request is granted, the latter generates and issues a pseudonym ID (PID_i) to the user.

$$PID_i \overset{\text{def}}{=} (psedd, ExpiryTime) \tag{4,51}$$

The same RID_i will be used in the group initialization as well as the key establishment process. The HSS also establishes and configures key parameters necessary for authenticating any formed MTC groups. Specifically, it generates a set of random numbers $\mathbf{R_z} \in \mathbf{Z_p^*}(z = 1,2,...i)$ and uses the set to compute a set of temporary identities $TID_{MTCD_{i-j}}$ to each $MTCD_{i-j}$ in a group:

$$TID_z = h_1(ID_{MTCD} \| \mathbf{R_z} * x) \tag{4,52}$$

where $h_1(.)$ is a secure hash function and x is HSS's secret key.

The HSS ultimately organizes the MTCD group into a binary tree [61]. Each node of the tree has a secret key that is known to each member MTCD. However, the secret keys of the nodes forming a path between a given MTCD and the root of the tree is not disclosed.

The HSS calculates a group key as follows:

$$GK_i = h_3(\sec_{i-1} \oplus \sec_{i-2} \oplus ..., \oplus \sec_{i-j} \oplus g * x) \tag{4,53}$$

where $g-$ is a random number, and $h_3(.)$ is a key generation function.

HSS further selects three hash functions; $h_1(.)$, $h_2(.)$, $h_3(.)$ which the key generation center (KGC) uses to generate an authentication message S to be used for group authentication. It also generates **k** tokens, all being a function of TID_{MTCD_i} each device. These tokens must remain secret to any device outside the group.

Finally, the KGC computes and publishes the hash, function of S, $H(S)$ as well as hash function $H(.)$ that will be used to verify the validity of all MTCDs in the group.

B: MTCD Group Authentication and Key Agreement

This commences when a set of identified MTCDs within network coverage range request access so as part of a service/ application rendering. These are identified as a group

($MTCD_{grpi-j}$). We assume that within the group, the device with higher communication capability as well as battery reserve will be designated as group leader ($grp_{i-leader}$). The service provider then assigns a key (K_{grpi-j}) to each group member, as well as generates a group key that will be shared by both the MTCD group and HSS. The group key is used by individual MTCDs in the group for mutual authentication as well as privacy protection between MTCDs and service providers.

This is carried out mainly by the MTCDs' group leader and the HSS where the user is located. This is accomplished in the following sequence:

1. Each MTCD group member broadcasts a fresh temporary identifier $TID_{MTCD_{i-j}}$ and associated token $f(TID_{MTCD_{i-j}})$ to the group leader.

$$MTCD_{i-j} \rightarrow \lfloor TID_{MTCD_{i-j}}, f(TID_{MTCD_{i-j}} \rfloor \Rightarrow MTCD_{i-leader} \tag{4,54}$$

2. The group leader computes the Lagrange component vector for the group (LC_{MMI}) using $TID_{MTCD_{i-j}}$ $f(TID_{MTCD_{i-j}})$ values received from the KGC, i.e $MTCD_{i-j} \rightarrow LC_{grpi}$.

The general formula used is:

$$LC_{grp} = f(TID_{MTCD_{1-i}}) \prod_{q=1, q\neq j}^{\frac{n}{m}} \frac{-TID_{MTCD_{1-q}}}{TID_{MTCD_{i-j}} - TID_{MTCD_{i-q}}} \bmod p \tag{4,55}$$

This component is broadcast back to all group members. Each member uses it to verify whether all members are legitimate, by calculating the secret key S and comparing the result with $H(S)$ published by the KGC during the registration phase.

3. The group leader further authenticates the group with the MME. In so doing, it first computes the group's $MACgr$ and $Auth_{grp_i}$.

$$MAC_{grp_i} = h_2(GK \| ID_{grp_i} \| LAI \| S') \tag{4,56}$$

$$Auth_{grp_i} = (TID_{grp_i} \| MAC_{grp_i}) \tag{4,57}$$

$$MTCD_{grp_i-leader} \xrightarrow{Auth_{grp_i}, TID_{MTCD_{-1}}, ...TID_{MTCD_{-j}}} MME \tag{4,58}$$

4. The MME confirms with the corresponding HSS whether the MTCD group is legitimate or not.

$$MME \rightarrow \xrightarrow{Auth_{grp_i}, LAI} HSS \tag{4,59}$$

5. Upon receipt of the authentication verification request message from MME, the HSS authenticates the group by computing the group's MAC_{grp_i} using values received from the MME versus those it has in store.

$$MAC'_{grp_i} = h_2(GK \| ID_{grp_i} \| LAI \| S) \tag{4,60}$$

$MAC'_{grp_i} = MAC_{grp_i}$ implies successful authentication by the HSS, and the MTCD group leader will be informed accordingly.

HSS also further generates a temporary group key GTK for the MTCG Group.

$$GTK_{grp_i} = h_3(GK \| r_{HSS}) \tag{4,61}$$

Where, r_{HSS} is a random number. It also generates a token to MME that will enable the devices to authenticate the MME in future sessions

$$HSS \rightarrow \xrightarrow{f(ID_{MME}) \| GTK_{grp_i} \| r_{HSS} r} MME \tag{4,62}$$

6. Upon receipt of messages from HSS, MME calculates its own Lagrange component LC_{MME} as well as $Auth_{MME}$ and broadcasts them to the MTCD group leader.

$$LC_{MME} = f(ID_{MME}) \prod_{q=1}^{\frac{n}{m}} \frac{-TID_{MTCD_{i-q}}}{ID_{MME} - TID_{MTCD_{i-q}}} \times \bmod p \tag{4,63}$$

$$Auth_{MME} = (LC_{MME} \| r_{MME} \oplus GTK \| r_{HSS} \| ID_{MME}) \tag{4,64}$$

Upon receiving $Auth_{MME}$, and encrypted KID_i the group leader broadcasts them to the rest of the group members.

7. Upon receiving the messages relayed from the MME each device updates its Lagrange component as follows:

$$LC\_new_{MTCD_{i-j}} = LC_{MTCD_{i-j}} * \frac{-ID_{MME}}{TID_{MTCD_{i-j}} - ID_{MME}} \tag{4,65}$$

Each device also uses the received r_{HSS} value to calculate GTK :

$$GTK_{grp_i} = h_3(GK \| r_{HSS}) \tag{4,66}$$

It also computes its integrity and cipher keys as well as $K_{asme}^{'MTCD_{grp_{i-j}}}$:

$$IK'_{grp_{i-j}} = h_4(ID_{grp_i} \| r_{HSS}) K_{grp_{i-j}} \tag{4,67}$$

$$CK'_{grp_{i-j}} = h_5(ID_{grp_i} \| r_{HSS}) K_{grp_{i-j}} \tag{4,68}$$

$$K_{asme}^{'MTCD_{grpi}} = KDF(GTK_{grpi} \| IK'_{grp_{i-j}} \| CK'_{grp_{i-ij}} \| ID_{grp_i} \| IMSI_{grp_{i-j}}) \tag{4,69}$$

It further computes its response value and sends it to the group leader.

$$XMAC'_{MTCD_{grp_{i-j}}} = h_1(ID_{grp_i} \| r_{HSS} \| IMSI_{grp_{i-j}})_{GTK_{grp_i}}$$

(4,70)

The group leader uses the response values from each of the group members to finally compute the group response.

$$XMAC_{grp_i} = h_1(XMAC_{MTCD\,grp1} \oplus XMAC_{MTCD\,grp_{1-2}} \oplus \oplus,....XMAC_{MTCD\,grpn})_{GRPK_1} \quad (4,71)$$

The Group leader finally passes the group response ($XMAC_{grp}$) to the MME for final authentication of each MTCD.

C: MTCD Joining or Exiting

If an MTCD joins or vacates an already authenticated group, the secret S must be updated to avoid the old member to continue knowing the secret and to avoid new members discovering and exploiting previous secret values S. As illustrated in Figure 4.17 when an MTCD joins, a new group key is generated:

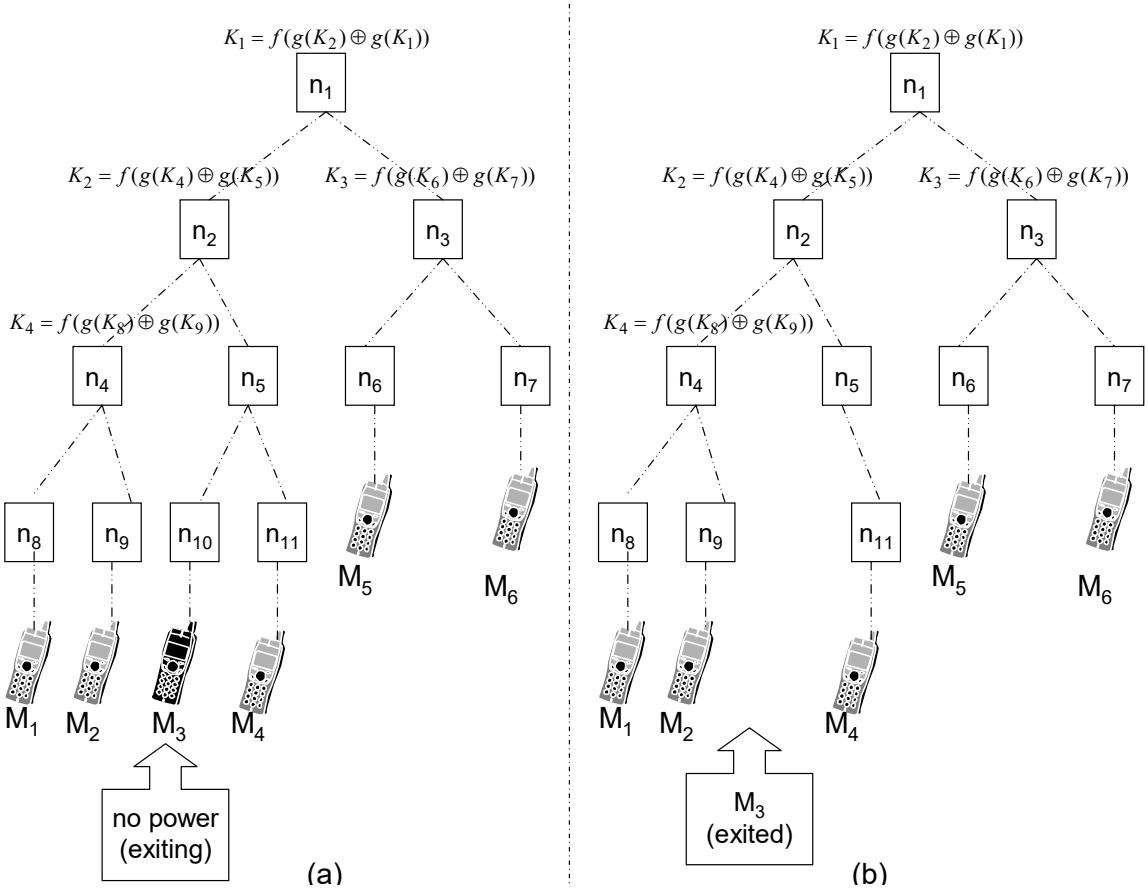


Figure 4.17: Example MTCD join/exit event tree

$$GK_i^{'} = h_3(GK \oplus \sec_{i-j}) \quad (4,72)$$

Where $\sec_{i-j}$ is the secret value of the node to which the new MTCD is located. Likewise, HSS generates a new value S as follows:

$$S_{new}=S+\delta S, \quad (4,73)$$

Where δS is a random value S generated each time a member joins or exits.

When a member exits, a new group key is computed according to:

$$GK_i^{''} = GK \oplus \sec_{i-j} \quad (4,74)$$

4.7.2 Analysis

In this section, we provide a general security analysis of the Gr-AKA protocol. Firstly we discuss its general security capabilities. We then proceed to evaluate its performance. We test some of the aspects of the Gr-AKA protocol using the AVISPA tool.

4.7.3 Security Analysis

User's Privacy: At the registration phase, **t**he MTC User's identity is mapped to a pseudonym ID (PID_i), and thereafter the latter is used for authentication purposes rather than the real User's name. In this way the user's real identity is concealed hence privacy is guaranteed.

Mutual authentication: The Gr-AKA protocol provides robust mutual authentications between $User\_HSS$, as well as among the individual $MTCD_{grp_i}$ members. HSS authenticates the user by way of verifying MAC values computed using the User's credentials such as $RID$and PID. To authenticate HSS, the User checks the received $MAC^{''}$ from the MME and if they both match with the $XMAC^{'}$, then both MME and HSS are authenticated.

Similarly, HSS verifies and authenticates the $MTCD$ group by verifying their Lagrange components. Each member then uses these Lagrange components to compute the secret S and compares it with the same value that was sent from the KGC .

Backward /Forward Key Secrecy. With the Gr-AKA protocol, the group key (GK) is updated and changed each time a device leaves or joins the group. When a device joins the

group, HSS is compelled to broadcast its secret node, thus a new GK is computed using equation (4.72).

Similarly, when a device exits, the remaining devices are compelled to update their GK using equation (4.74).

Attack resistivity: The channel between the MTC user and MTCD group is open to various attacks. To safeguard against replay attacks, time-stamped key hint messages are periodically exchanged between the two parties. A hacker who successfully intercepts the key hints exchange will not be able to replay a message for the next key hint exchange message because of the time stamping.

MiTM attack: The channel between the MME and HSS is assumed to be secure (in terms of integrity, confidentiality, and entity authentication), and only the channel between $MTCD_{grp_{i-j}}$ and MME may be vulnerable to MiTM attacks. However, in the proposed protocol, the use of Shamir's secret sharing [35], together with the Lagrange component, makes it extremely difficult to recover the secret token. Furthermore, the group's ID is secret thus further making it difficult for attackers to generate or verify the MAC_{grp_i}.

4.7.4 Performance Analysis

The protocol is analyzed in terms of its general security capabilities, computational demands/complexity as well as signaling overheads. The main security aspects of the protocol are tested using the Automated Validation of the Internet Security Protocols and Applications (AVISPA) tool [41]. We first created the model as in Figure 4.17, and also specified the basic roles. We were able to verify that it can guarantee the privacy of a generated session key, as well as general authentication between MME and HSS.

To evaluate the total computational overheads, we compare the protocol to similar proposed protocols such as PPAKA-HMAC [55], G-AKA [55], and GBS-AKA [56]. In our analysis, we assume that overall there are n $MTCDs$ and each can have up to m members. The following cryptographic computational times are utilized; map to point hash operation ($T_{mtp}=0.07ms$), $MTCD$ Lagrange component computational time ($T_{L-MTCD}=0.06ms$), HSS Lagrange computational time ($T_{L-HSS}=0.04ms$), multiplication over an elliptical curving ($T_{mul}=0.6ms$), pairing

(T_{pair}=4.5ms), hash operation (T_{hash}=0.07ms), symmetric ciphering/deciphering (T_{aes}=0.16ms),. Table 4.9, summarizes the computational overheads of the 4 protocols.

Table 4.9: Computation complexity of Group Protocols

AKA Protocol	Computational overhead		
	MTC Devices	Network	Total (sum)
PPAKA-EXAC [12]	$3T_{hash}*n-(T_{hash})*m$	$2T_{hash}*n-(T_{hash})*m$	$5T_{hash}*n-(2T_{hash})*m$ $2*Rand-(n-10)$ $*\exp(5-n)$ $*H_{var}-(2n-1)$ $*Mul-1*Hash$
G-AKA[13]	$4T_{exp}-(4T_{exp})(n-1)$	$3(T_{exp})*n*(2T_{exp}h)*m$	$7T_{exp}*n-(2T_{exp})*m$
GBS-AKA [14]	$(T_{exp}-2T_{exp})*n-2T_{exp}*m$	$(T_{exp}-2T_{exp})*n$ $-(6T_{exp}-T_{enc})*m$	$(2T_{exp}-4T_{exp})*n$ $-(8T_{exp}-T_{enc})*m$
GR-AKA (proposed)	$(4T_{exp}-2T_{enc})*n$ $-(2T_{exp})*m$	$(3T_{exp}-2T_{enc})*n$ $-(2T_{exp})*m$	$(7T_{exp}-4T_{enc})*n$ $-(4T_{exp})*m$

The table above was adopted from [56].We explored execution key generation time as a function of key size in bits. The key size is varied from 2^1 to about 2^{13} bits.

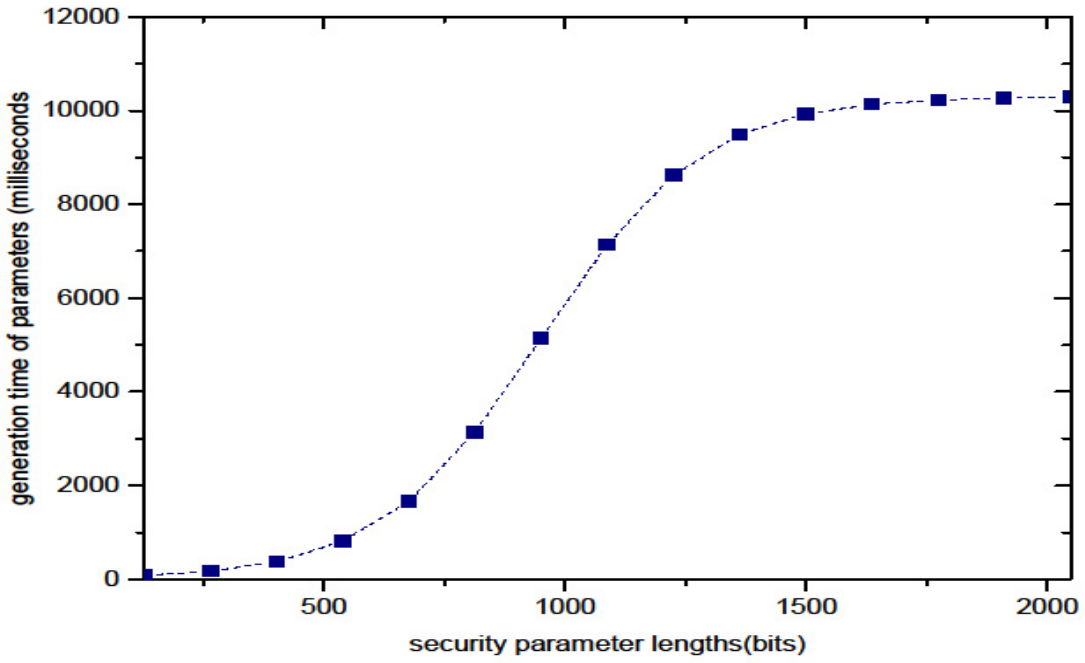


Figure 4.18: Overall protocol key generation time versus size

From the plot in Figure.4.18, it is observed that as the key size is increased, so does the key generation time. However, increasing the key length makes it more secure. We thus chose to

fix the key size to 12 bits (2^{11}), which is a 7 seconds delay. This is not so much a hindrance as key generation is a once-off operation during initialization. We also evaluate the proposed protocol's execution time.

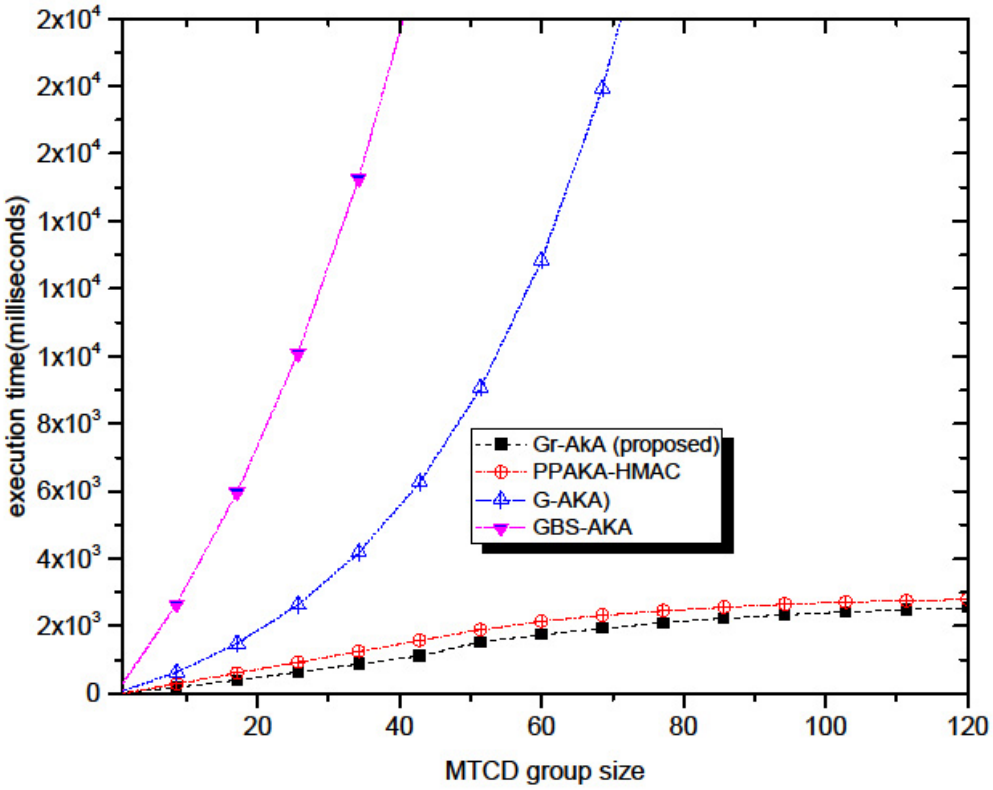


Figure 4.19: Execution time comparisons

The execution time increases linearly with an increase in MTCD group size for the proposed protocol as well as PPAKA-HMAC [57]. However, we see an exponential increase in execution times with the other two protocols as indicated in the same graph (Figure 4.19). We also analyze the overall magnitude signaling (communication) overheads of the proposed protocol. Overall the total signaling bits are computed from all the messages exchanged during the authentication process. Figure 4.20 shows plots of total signaling (communication) overhead as a function of the number of MTCD groups, each comprising 5 members. The proposed protocol together with the PPAKA-HMAC generate more or less the same levels of signaling data and is not so excessive to cause congestion in the signaling channels.

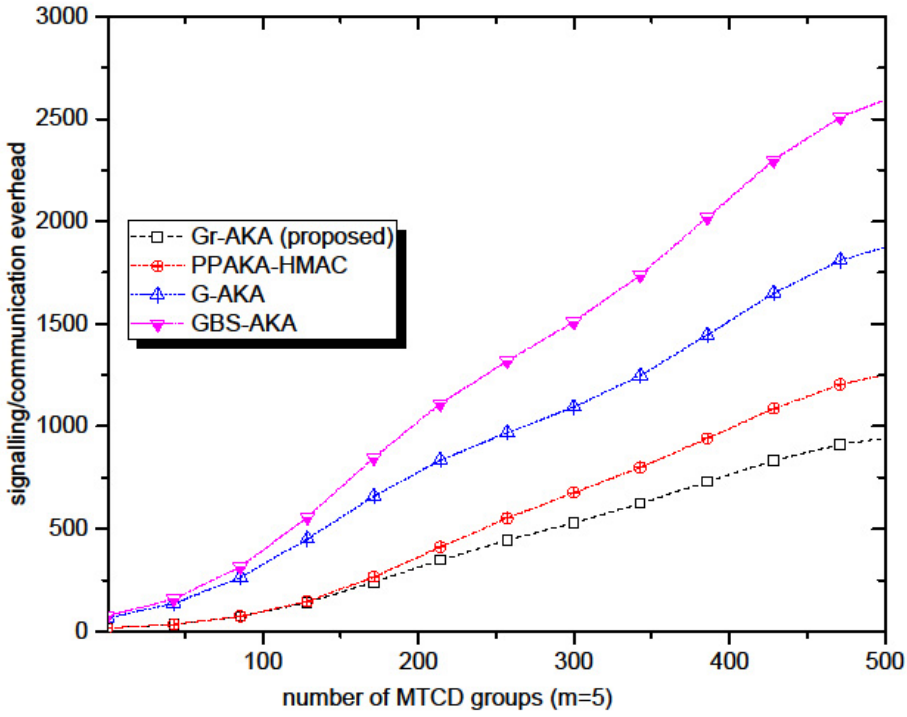


Figure 4.20: Signalling overhead

4.8 Summary Chapter Conclusions

D2D based technologies focus on facilitating direct linkage and communications among any communicating capable devices, without the mediation of the base traditional transport network infrastructure. In this regard, essential security objectives required for the cohesive interaction among the D2D communicating devices would include, primitives such as the preservation of integrity, confidentiality, robustness, resilience, and availability in case of any intentional or unintended intrusion attacks. This chapter explores a few protocols in terms of their security capabilities as well as computational complexities in terms of overheads. The development of the 5th generation of mobile networks is directly related to the IoT, hence, D2D communication, which provides direct communication between two devices without the intermediation of network infrastructure, as the 3GPP core network. Since D2D is still in its early stages, some concerns must be considered for its full implementation. A few D2D communication-based authentication schemes were reviewed in this chapter.

The review centers on both security and general performance. The fact that most IoT based devices are resources constraint implies that the current, as well as traditional authentication and key agreement, cannot be applied directly in fulfilling D2D authentication, The resource-constrained nature of the devices means that essentially we should rather organize devices in groups in which a group leader performs authentication on behalf of its members.

The group leader is selected based on the criteria such as computational power capability, storage, capacity battery life longevity as well as accessibility by all members Because the group leader receives basic authentication parameters and signatures from all devices in the group and aggregates them into one single signature, which enables the authentication of a group of devices in a single bilinear pairing operation, computational power, communication overheads as well as energy consumption are kept at a minimum.

5. A Lightweight Encryption Based Privacy and Security Framework

5.1 Introduction

Several group privacy and security protocols specifically relating to groups AKA are being explored. Security requirements such as confidentiality, mutual authentication, privacy preservation, integrity and most importantly utilizing a common and single security (encryption) key during the communication sessions in the IoT network is preferred. Such protocols need to inherently achieve efficacy in maintaining the group key unlinkability as well as generate minimal overheads that otherwise may lead to network congestion [59]. To alleviate signaling-related congestion the authors in [60] proposed a congestion avoidance approach in which a group of devices delegates a leader to handle the communications on behalf of the rest of the group members. In this way, the volumes of aggregated signaling overheads are significantly lowered and so is the congestion.

The same approach was revisited by the authors in [61] in which they propose a group AKA (G-AKA) protocol. In this case, a single device from the group is authenticated by the AKA authority in the SG, after which the same device is now delegated to authenticate the remaining devices of the group. In that way, the authentication process becomes relatively simplified for the rest of the devices in the group. One disadvantage with such a protocol is that of the possibility of high levels of signaling overhead being generated should several devices wish to gain access to the SG network simultaneously. It has also been shown that the protocol is so secure in preventing potential threats such, as DoS and redirection attacks.

Asymmetric key-based AKA (SE-AKA) protocol that enhances both data integrity and confidentiality was investigated in [62]. Whereas the protocol shows security improvements, it generates massive signaling overheads that ultimately lead to network signaling congestion.

In [63] an enhanced group AKA (EG-AKA) protocol is proposed to authenticate a targeted group of devices. The protocol is quite computationally intensive and hence generates high computation overheads in the network due to asymmetric key operations. The authors in [64] propose a Group-AKA protocol that mitigates the problem of excessive signaling overheads by way of authenticating grouped devices simultaneously. The protocol maintains the unlinkability in the group key whenever an individual device vacates or joins the group. One of its

shortcomings is that of preserving the privacy of participating devices as well as susceptibility to identity-catching attacks while authenticating any additional new device(s) into the group.

To address the shortcomings of privacy preservation failures in previous AKA protocols, the authors in [65] proposed elliptic curve cryptography-based privacy-preserving group authentication AKA (PRIVACY-AKA). Initially, a pseudo-identity by way of elliptic curve cryptography is generated and thereafter each device in the group transfers its message authentication code to the designated group leader. The group leader then in turn compiles each code into an aggregate MAC which will subsequently be used by the network to authenticate the rest of the devices in the group. Whereas the protocol provides acceptable security, it, however, generates high computational overhead due to the asymmetric key cryptosystem. It also fails to take into consideration the group's key secrecy in terms of when a device joins or vacates the group.

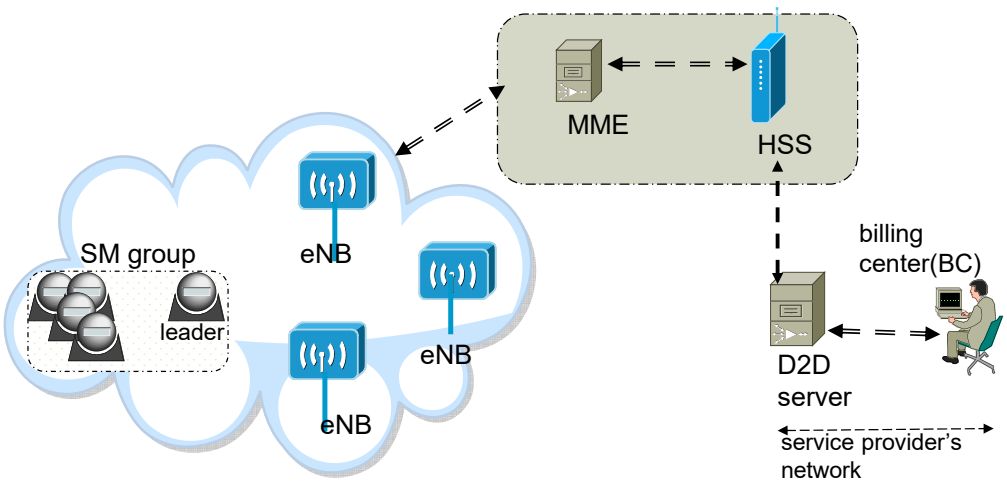


Figure 5.1: 3GPP coverage in an IoT network

The proposed approach overcomes the security problems of the network and generates relatively less overhead compared to the existing group-based AKA protocols. It accomplishes all the security requirements for D2D communication with moderate levels of both signalings as well as computational overhead.

To provide privacy as well as security in surveillance secure, secure authentication and key exchange among the D2D communication compliant smart surveillance cameras is necessary. At the local level, a 3GPP IoT-enabled network architecture, as well as coverage, is assumed as illustrated in Figure 5.1. Key security-related blocks defining such a basic infrastructure will incorporate a D2D communication server, (D2D), home subscriber server (HSS), and

mobility management entity (MME). The HSS retains the attributes information of the surveillance cameras (devices) and relies on the MME to verify each unit by way of granting a set of authentication tokens.

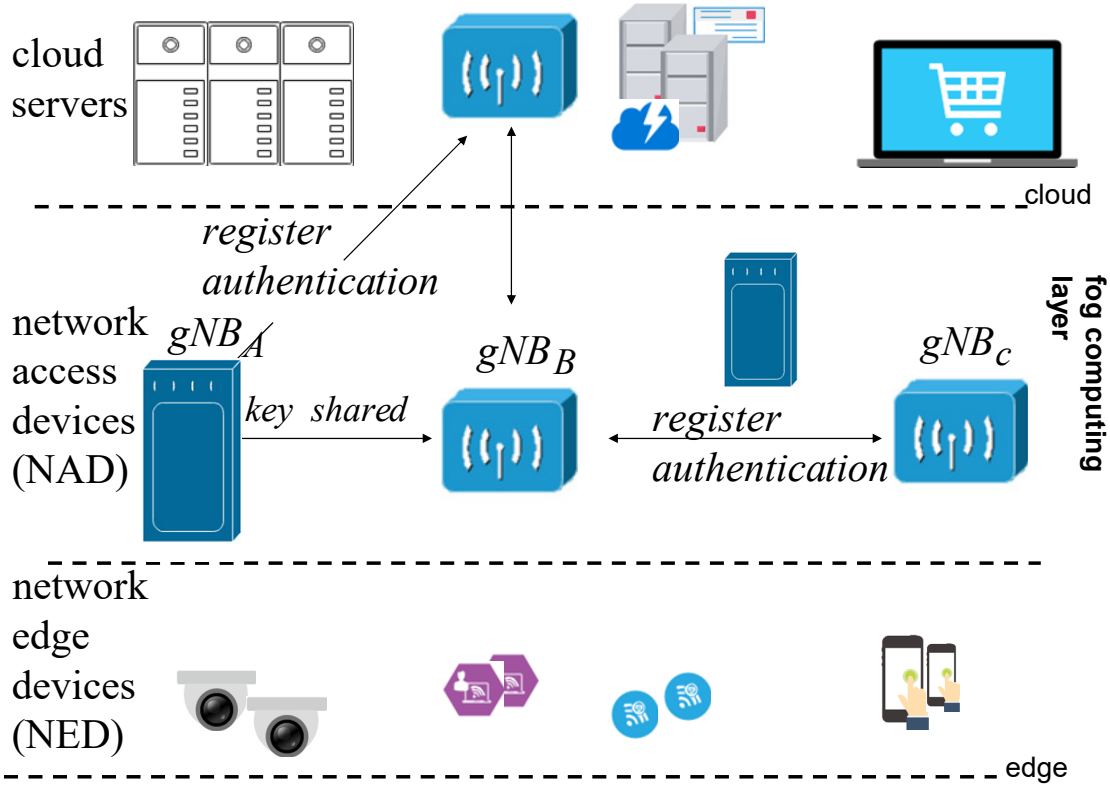


Figure 5.2: Fog Computing paradigm Alternative,[66]

The base infrastructure just discussed cannot guarantee a high level of QoS and this is because of the limited computing (processing) capabilities of the devices themselves. The cloud computing paradigm is also being explored as an alternative. This is because it can render a better QoS to users with elastic resources despite its limitations. The key limitation is that of long round-trip times. Besides, it cannot cope up with the low latency requirements of most human directly related services and applications. Hence of recent the Fog Computing paradigm was introduced to directly respond to the latency minimization issue. It exploits the fog layer, which is the interfacing layer between the core and peripheral network sections to drastically reduce latencies as well as boost the limited computing powers in resource-constrained devices. It can also provide network context information which ultimately is used by fog applications and services to optimize context-awareness. Its support for location-awareness; means it can fully support device mobility which is a direct booster for location-based services and applications. Fog computing easily provides a local overview whereas a global overview will

still be provided by cloud computing. Primarily a fog computing model comprises key elements such as (i) network edge device (NED), (ii) network access device (NAD), i.e., fog node, in the proximity of a NAD, and (iii) cloud server (CS).

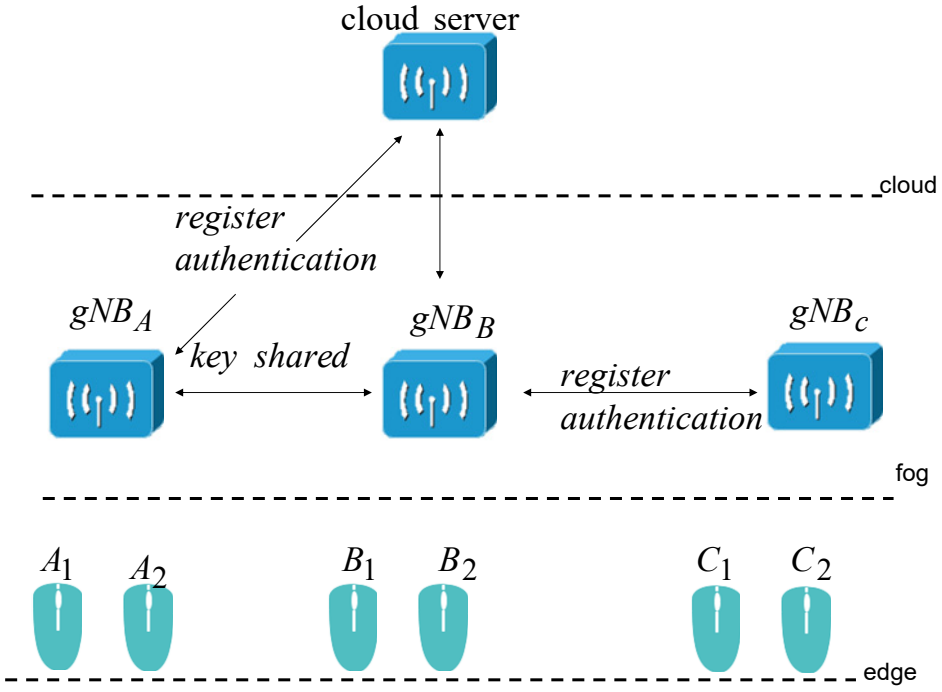


Figure 5.3: Authentication delegation at Fog layer

The NEDs will mostly be populated by various deice-constrained devices such as micro-powered smart devices and sensors acquiring data in a specified locality. The NAD is enhanced with more computation capabilities, and given that it has a more reliable power supply, it can thus be bestowed with authentication functionalities as it will always be available.

5.2 Proposed Security Framework

The data exchanges between the various entities constituting the AMI traverse one or multiple collectors and possibly through other SMs acting as relay points. D2D communications is assumed between the BC and DCs. As such all SMs deployed in the SG are assumed to be D2D communication compliant and physically unclonable. Data load handling in SMs is addressed by way of data aggregation in which the data from various remote SMs is combined together before being relayed across the network via a designated relaying SM. The same relaying SM becomes a group leader (SM_{gl}). In that way, both the bandwidth as well as links are utilized more efficiently.

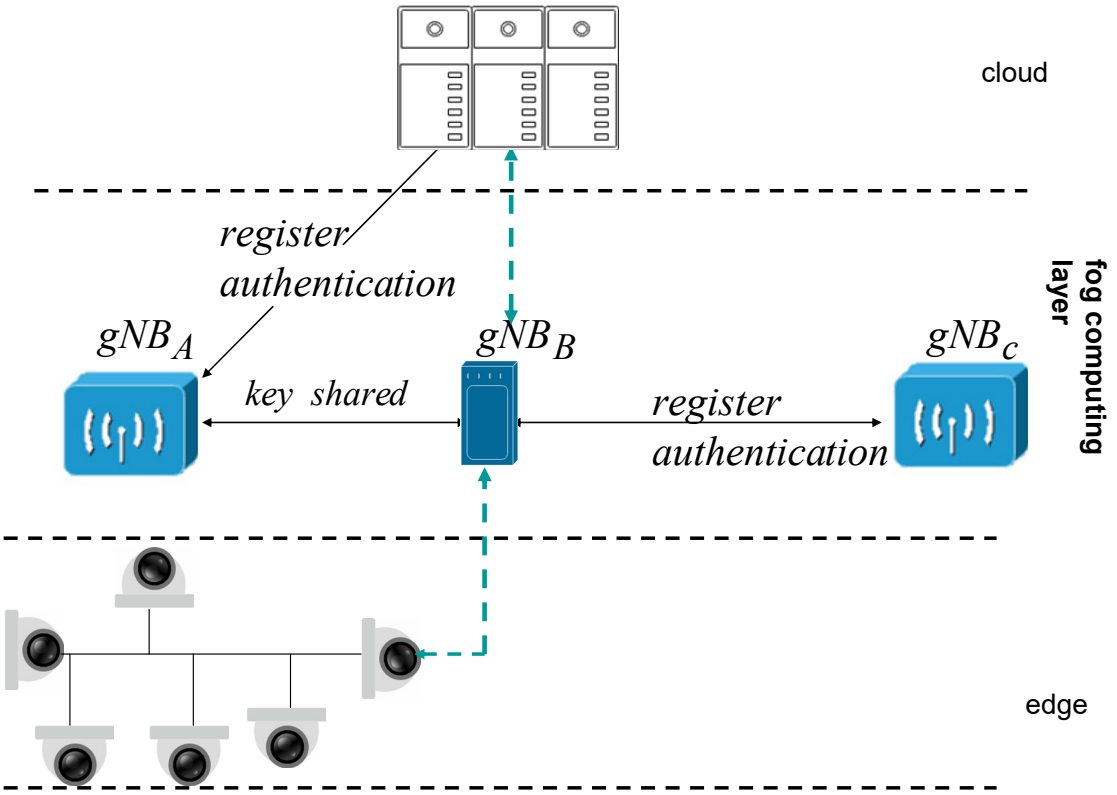


Figure 5.4: System model [67]

The surveillance service will involve several surveillance cameras deployed in a particular target area(s). It is important to ensure that both security and high-level privacy are maintained throughout. By default, the service will involve secure authentication as well as key agreement and exchange among the D2D communication compliant smart surveillance cameras. A 3GPP or equivalent authentication network architecture is assumed as illustrated in Fig. 5.4. Key security-related blocks are the CS, fog computing layer as well as edge network. Data exchanges between the various entities constituting the service may traverse several intermediate relay units. The system exploits the fog computing layer to reduce any undesired end-to-end latencies due to constraint computing resources within the devices. Besides, the Fog layer has several entities such as gNBs and other wireless access points hence it is necessary that within this layer level, interaction among the entities is facilitated so that loads can be evened.

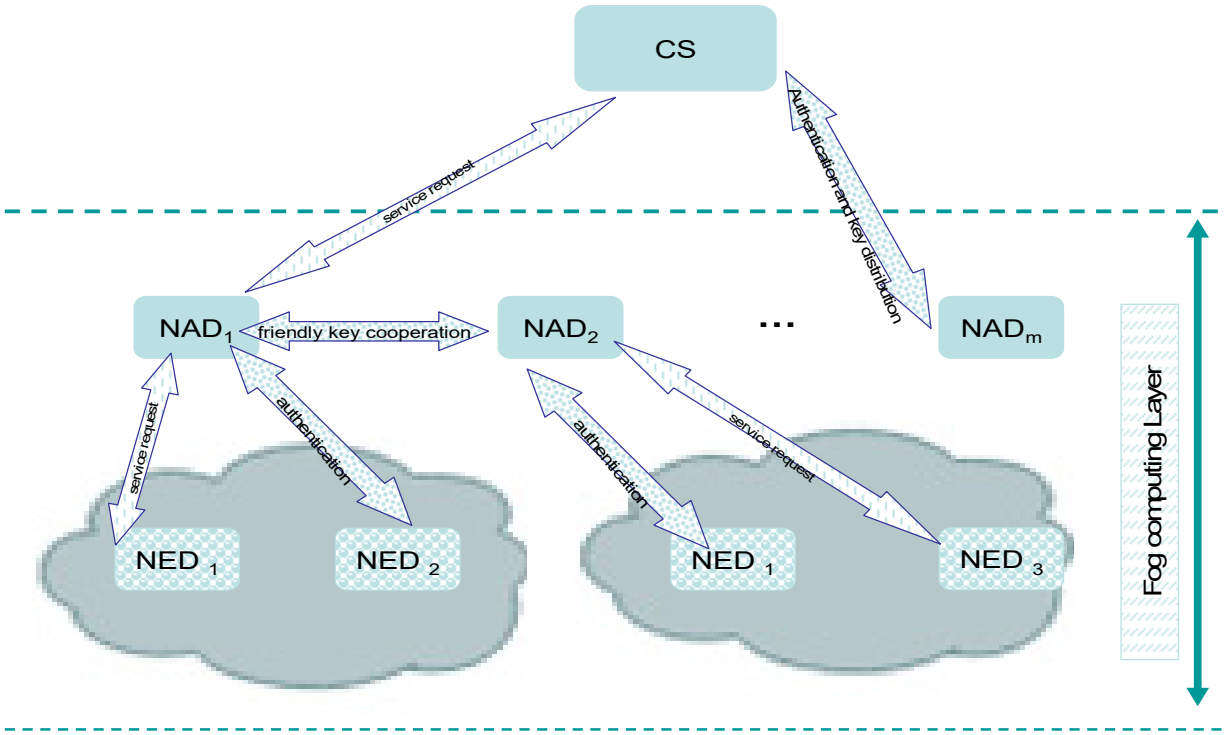


Figure 5.5: D2D aided Fog Computing, [66]

Figure 5.5 illustrates a D2D aided Fog computing model example. Several authentication scenarios are supported. In case the mobile smart surveillance cameras are mobile and it moves to a new NAD, then the most recently attached NAD will assist in the authentication process.

5.3 Service Authentication details

In this section, we briefly described the key steps of this service authentication scheme whose aim is to ensure anonymity, user un-traceability, backward as well as forward secrecy, authentication key agreement, as well as mutual authentication. The scheme comprises three key phases namely; registration, authentication, and data exchange phases.

A: Registration and Service Authentication

A user will have to be registered and authenticated first before using the service. The service itself also comprises a group of smart surveillance cameras (SCs) that collaborate in rendering the service in a particular target area. Thus both the user and devices (SCs) will have to be authenticated first before the service can commence. This is done using the normal 3 GPP network infrastructural procedures. Because the SCs work as a group it would be appropriate for the work herein to advocate for a Group Authentication and Key agreement approach.

Each SC has two embedded keys; a private (A^{-}) as well as a public key (A^{+}) one. Similarly, the $gNBs$ are also equipped with private and public keys. If the need arises, a commonly shared key would normally be established using Diffie-Hellman (DH) parameters g and p [69]. In the process, some primary cryptographic operations have to be adhered to as follows:

- For a message that requires public encryption we have;

$$M \rightarrow Public\_key(K, M) \tag{5,1}$$

- To further enhance the security of the same message, we can symmetrically encrypt it as follows:

$$i \rightarrow symmetric\_key(K, M) \tag{5,2}$$

- To endorse a signature on the encrypted message we have;

$$signature)\_(A, M) \tag{5,3}$$

where, A is s a derivative of (A^{-}).

- The same key is utilized when calculating a one way hash function;

$$hash\_(k, M) \tag{5,4}$$

- We assume a centralized key generation center (KGC) and is available for use within the SG by authorized parties.

A primary security objective is that the images (data) captured from the SCs must be confidential and be only accessible to those who are granted authorization. All the SCs in the group relay their data (images) via the designated group leader. (SC_{gl}). The collected data is then forwarded to authorized users via the network.

The detailed descriptions are as follows [70], [71];

B: Session Initialization and SC Group Registration

We assume that a user wishes to subscribe to the video surveillance service.

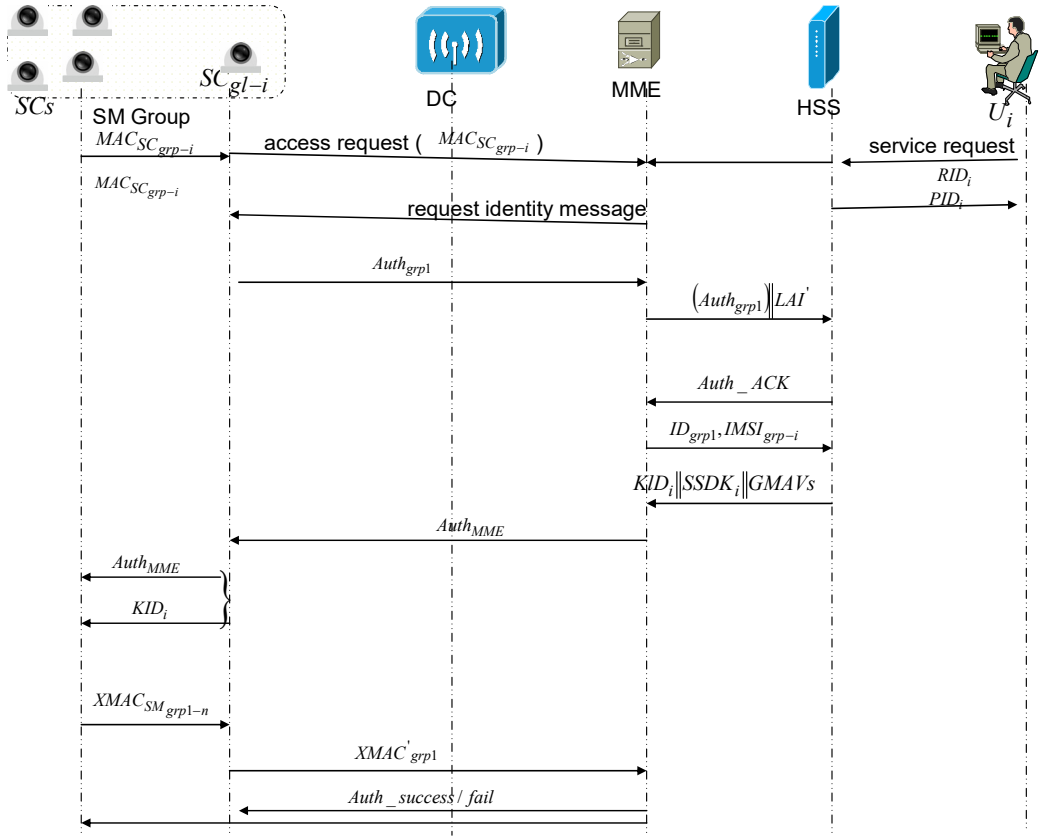


Figure 5.6: Summary events

The user will use his/her real identifier (RID_i) for the preliminary registration formalities with the local *HSS* . Upon confirmation of registration, the latter issues the user a pseudonym ID:

$$pseudo\_PID_i \overset{\text{def}}{=} (pseudoID, ExpiryTime) \tag{5,5}$$

The next task is to discover a surveillance smart camera group SC_{grp-i} and have all its members authenticated by the *HSS* as well. In this regard, the *HSS* randomly generates a set of real-valued integers ($\mathbf{R_z} \in \mathbf{Z_p^*}(z = 1,2,...i)$ that will be used to compute a set of temporary identities $TID_{SM_{i-j}}$ to be assigned to each of the members of the group.

$$TID_z = h_1(ID_{MTCD} \| \mathbf{R_z} * x) \tag{5,6}$$

The *HSS* further computes an authentication key for the group as follows:

$$GK_i = h_3(\text{sec}_{i-1} \oplus \text{sec}_{i-2} \oplus ..., \oplus \text{sec}_{i-j} \oplus g * x) \tag{5,7}$$

where $g -$ and $h_3(.)$ are a random integer. And a hash key respectively.

C: Group Authentication and Key Agreement

To maintain a trust relationship within the group, mutual authentication is mandatory. In this regard, the service provider issues a key ($K_{grp\text{-}i\text{-}j}$) to each group member for that purpose. However in order to keep communication costs to a minimum in the network, the designated group leader (SC_{gl-i}) will act as an intermediary for all members, hence all communications are relayed via it to the HSS . Key steps during this time will be as follows:

1. A fresh temporary identifier ($TID_{SM_{i-j}}$), as well as token, $f(TID_{SM_{i-j}})$ is shared by each group member and the leader.

$$SC_{i-j} \rightarrow \left[TID_{SM_{i-j}}, f(TID_{SM_{i-j}} \right] \Rightarrow SC_{gl} \tag{5,8}$$

2. A Lagrange component (LC) vector for the group computation follows. This will require $TID_{SM_{i-j}}$ and $f(TID_{SM_{i-j}})$ from the 3GPP network

$$LC_{grp-i} = f(TID_{SC_{1-i}}) \prod_{q=1, q\neq j}^{\frac{n}{m}} \frac{-TID_{SC_{1-q}}}{TID_{SC_{i-j}} - TID_{SC_{i-q}}} \bmod p \tag{5,9}$$

The LC will be shared with the group members, and they in turn utilize it for mutual authentication purposes within the group as well as the core network (MME). The authenticating with the network (MME).requires group's MAC_{grp-i} and $Auth_{grp-i}$ c.

$$MAC_{grp-i} = h_2(GK \| ID_{grp-i} \| LAI \| S') \tag{5,10}$$

$$Auth_{grp-i} = (TID_{grp-i} \| MAC_{grp-i}) \tag{5,11}$$

$$SM_{gl-i} \xrightarrow{Auth_{grp-i}, TID_{SM_{i-1}}, ..TID_{SM_{i-j}}} MME \tag{5,12}$$

We can extend this procedure to the Fog Computing paradigm which we defined and explained in subsection 5.1. In this case, both the registration and authentication phases are e detailed in Figures 5.7 and 5.8 respectively.

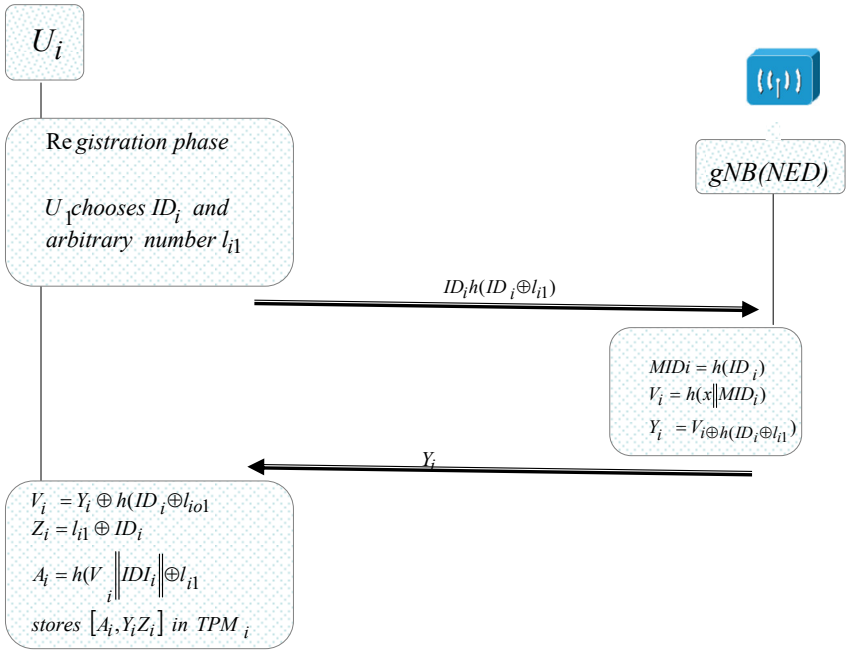


Figure 5.7: User Registration process

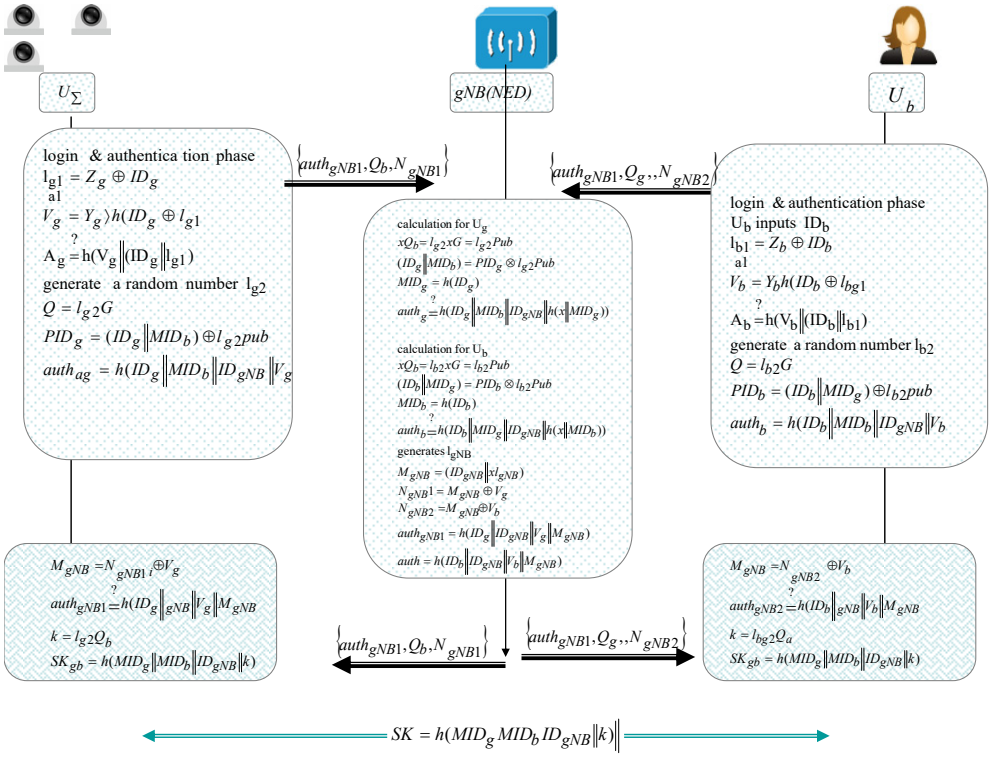


Figure 5.8: Authentication Phase

Table 5.1: Notations used and definition of symbols

symbol	*definition*
U_i	**subscriber to a service**
ID_i	U_i **'s ID**
IG_{gNB}	**Gateway ode's identity**
x	*gNB* **'s secret key**
PUb = xG	*gNB* **'s public key**
$E_p(e,f)$	**An elliptic curve**
G	**Base point of elliptic curve (** $E_p(e,f)$ **)**
$h(\cdot)$	**One way hash function**
$\oplus$	**Exclusive or operation**
$\|\|$	**Concatenation operator**

Note that overall, the *gNB* are key in establishing secure communication among devices and linking with the cloud server. Once a group (and *HSS*) are authenticated, they can now securely exchange data. The resource constraint factor still faces challenges. Once both the user and surveillance smart camera group have been authenticated, then the *gNB* dispatches a challenge message to both parties. Once they receive the message, each once again authenticates the *gNB* and ultimately computes a session key. In the end, both would have agreed on a common key .

5.4 Informal Security and Performance Analysis

In this section, we briefly analyze the performance of this framework. Overall it is deduced in the analysis that the framework has the potential to provide robust as well as resilience security to any authenticated.D2D communications-based service or application. Note that we provide an informal analysis and this is concerning its correctness as well as ensuring that it resists any attacks.

- Identity Security. The framework advocates for lightweight encryption solution approaches. This is because the associated key devices are natural resources constrained. This further integrates the existing resource constraint infrastructure with the fog-assisted computing paradigm and in that way, realistic security measures are implementable to ensure identity security.
- Mutual Authentication. In the framework, there are two possible authentication pathways, the normal 3GPP which can face constraints such as insufficient computing resources; as well as the fog layer assisted infrastructure, in which the authentication role is assigned to the gNB . Equations (5,10), (5,11) and (5,12) ensure mutual authentication. Moreover, the gNB separately authenticate the device group and user and in the process create a common shared key that will be used in the data exchange phase.
- User Anonymity. Pseudonym identities are used for both normal 3GPP and Fog layers assisted computing infrastructural approaches. Hence it becomes very difficult for hackers to decode the true identifies of the various parties participating in the established service. Note that the pseudonym Id is sent over public channels but the real ID is not. In that way, User anonymity is guaranteed.
- User untraceability. The creation and reliance thereafter of encrypted PIDs ensures that users and formed groups cannot be traced in both location as well as true ID. The fact that the PID (which is a dynamic ID) is used implies that tracing is near impossible for attackers.
- Forward Secrecy. In the unlikely event that an attacker gets hold of parameters such as the secret key of a HSS or gNB , he cannot determine the former session keys. This is because random numbers are used in the generation of such keys, and thus keys change with every new session.

- Backward Secrecy. It follows from the principle used in Forward secrecy that even if the secret key of a gNB was accidentally compromised, future session keys cannot be determined.
- Replay Attack. The framework is immune to compromise by replay attacks in that in the unlikely event that a request message is accidentally intercepted, replaying will not materialize as random numbers are used for every new session.

5.5 Fog Computing Based Lightweight Authentication Protocol

Under this section, we apply the fog computing paradigm-based framework principles discussed in the previous section to describe and analyze a lightweight authentication protocol for a surveillance service. In so doing our focus is on both the protocol's security resilience and robustness, communication, computational as well as energy efficiencies. In short, any authorized (authenticated D2D communication compliant surveillance camera and the integral service as a whole should resist any attacks.

It should be noted that lightweight encryption is advocated for as the surveillance cameras do not incorporate enough computing and power storage capabilities hence categorized as resource-constrained. The Fog computing infrastructure will also assist in reducing the end-to-end latencies i.e between the user who subscribes to the service and the targeted surveillance cameras.

Both Identity security and mutual authentication should also be guaranteed. Because we have incorporated the Fog computing layer, the mutual authentication is effectively enhanced by the existence of two pathways for its execution, the existing 3GPP and the fog layer assisted infrastructure.

Pseudonym identities will be used for both the subscribe (user) r to the service and the individual surveillance cameras, which are deployed in groups of varying sizes, depending on coverage area. The use of pseudonyms makes it problematic for hackers to decode the real identities of the parties involved. To ensure seamless identity security, the pseudonym identity is exchanged over public channels whereas the real identities are not. Pseudonym identities are used for both normal 3GPP and Fog layers assisted computing infrastructural approaches. Hence it becomes very difficult for hackers to decode the true identifies of the various parties participating in the established service. Note that the pseudonym Id is sent over public channels but the real ID exchange is restricted to encrypted D2D channels.

5.5.1 Initial Service Registration

Because of security threats in a particular region, several varying size groups of surveillance cameras (edge devices) are deployed. Each surveillance camera or edge device has to register with the cloud computing service (CCS) via a secured link. It will generally follow the following procedural steps:

ED_i formalizes a request to be registered with the CCS.

CCS imitates and n - bit counter $gcount$ which will be automatically incremented for each formal request received.

CSS increments $gcount$, i.e. $[gcount]+1$, computing a transaction sequence number, $T_{seq}=\{gcount\}+1$, a secret key K_{ec}, and a pseudo ID $PID = \{pid_1, pid_{2,....pid_n}\}$ that are assumed unlinkable.

The CSS dispatches the parameters generated in the previous steps together with a group key GK to the ED_i

5.5.2 Authentication with Fog layer

This takes place when for the first time a member in a group wishes to exchange data (mages captured) to the CCS. . The procedural steps can be summarised as follows:

ED_i contacts the nearest network access device (NAD) and furnishes it with:

$$ED_i \rightarrow NAD : M_{A_1}: \{AID, N_x, T_{seq}\} \quad (5.13)$$

The information is generated as follows: $N_x = N_e \oplus K_{ec}$ is computed by ED_i, where, where N_e is a randomly generated number. Similarly ED_i generates $AID = h(ID_{ED_i} \| K_{ec} \| T_{seq})$ and ID_{ED_i} is the surveillance camera's ID. K_{ec} is computed from any one of the unused pid s i.e

$$K_{ec} = AID = pid_j, k_{em_j} \quad (5.14)$$

Because at this stage the two parties are unknown to each other, this information (request message) will be rerouted to the CCS.

$$NAD \rightarrow CCS : M_{A_2} : \left\{ Fwd, M_{A_1} \right\} \tag{5.15}$$

Upon reception of the message M_{A_2} from NAD, it verifies this information. This it carries out as follows: Firstly it locates the T_{seq} from the local database (DB) and it turn retrieves ID_{ED_i} as well as K_{ec} from the same local DB to use them for the verification process. If verification succeeds, the CSS generates a communication key CK and a new one $T_{seq_{new}}$. Ultimately the surveillance camera (ED_i)the following:

$$e1 = k(K_{ec} \| T_{seq}) \oplus T_{seq_{new}} \tag{5.16}$$

$$e2 = h(K_{ec} \| \ ID_{ED_i}) \oplus CK \tag{5.17}$$

and

$$\text{Res}_{CCS} = h(e1 \ \| e2 \ \| K_{ec}) \tag{5.18}$$

as well as updating;

$$T_{seq} = T_{seq_{new}} \tag{5.19}$$

CCS then confirms all this to the NAD in the form of a response message M_{A_3}

Upon receiving the confirmation message M_{A_3} from CCS, the NAD accordingly generates a tracking number, $Track$ No. as well as a random number R_n before computing;

$$\text{TN} = \text{h}(\text{CK} \| \text{R}_\text{n}) \oplus Track \ \text{No.} \tag{5.20}$$

and

$$\text{Res}_{\text{NAF}} = h(Track \ \ No. \| CK \ \| R_n) \tag{5.21}$$

It then sends a confirmation message M_{A_4} to the surveillance camera ED_i.

Once ED_1, receives the message M_{A_4}, it will verify the validity of the response parameters Res_{CCS} and Res_{NAL} before decoding $T_{seq_{new}}$ and CK. Ultimately it will also update T_{seq} to $T_{seq_{new}}$.

Because in D2D fog-assisted computing, neighboring devices (i.e. in a group) can assist one another by secluding any outsider (hackers). In this case, they will share a channel (link) key K_{ij}. In this case, the authentication process can be summarised as follows:

When it becomes necessary for another surveillance camera ED_j to liaise with a NAD, then the NAD can authenticate it with the help of the most recently authenticated device in this case ED_1. In this case, ED_j furnishes its identity as an alias identity:

$$AID = h(ID_{ED_j}, \| GK \| T_{seq}) \tag{5.22}$$

as well as generating a common group authentication request;

$$G_{auth} = h(ID_{ED_j} \| R_n \| GK \| K_{ij}) \tag{5.23}$$

Once ED_1 has received a request message M_{B_1} from ED_j it will carry out the necessary verifications before sending a confirmation message M_{B_2} to the NAD.

Upon reception of the confirmation message M_{B_2} from ED_1 the NAD validates all key parameters including the Track number($TrackNo$.), and also decodes tk . After successful validation of all key parameters, it will send a response message M_{B_4} to ED_1.

Upon receiving M_{B_4} from NAD, it checks the validity of Res_{NAL} as well as encoding the tk key. The latter is done using both the link key (K_{ij}) and the group key (KC)

$$tk^{\#} = h(GK \| ID_{ED_j} \| K_{ij}) \oplus tk \tag{5.24}$$

Ultimately it sends a confirmation message M_{B_4} to ED_j .

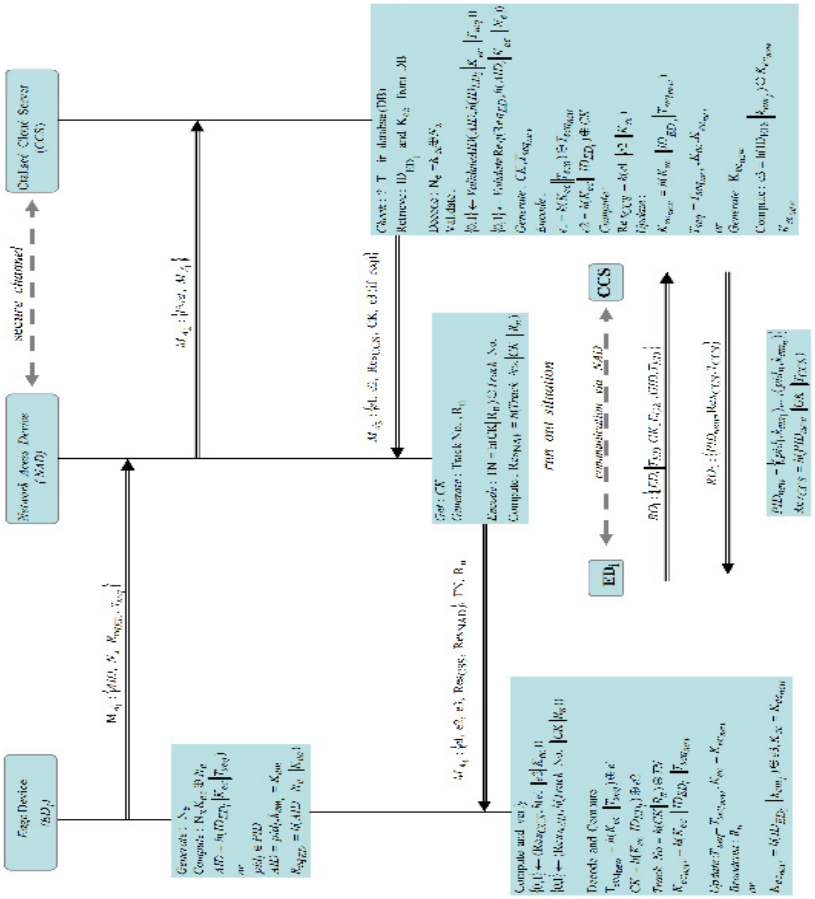


Figure 5.9: Initial Authentication for D2D-Aided Fog Computing

After receiving M_{B_4}, ED_j validates it before broadcasting a random number Rn to all group members. This is necessary to protect the group from replay attacks. The key steps are summarised in the next figure.

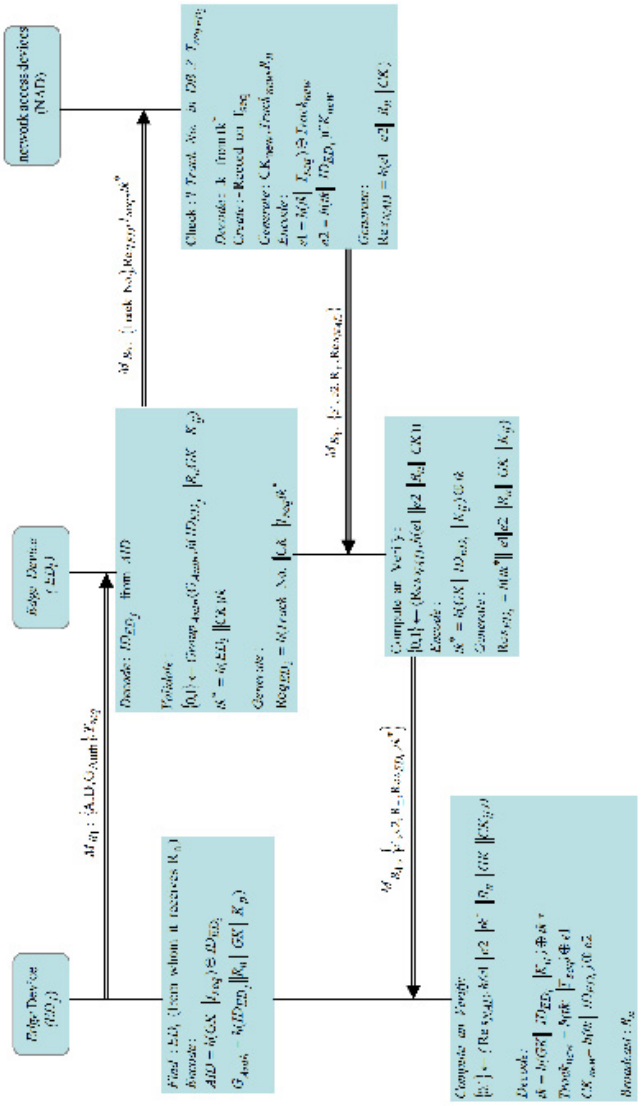


Figure 5.10. Example Authentication when are EDs are in cooperation

When a surveillance camera is moved to a new NAD, then the last visited NAD will assist the new one in verifying it. This is summarised in the next Figure.

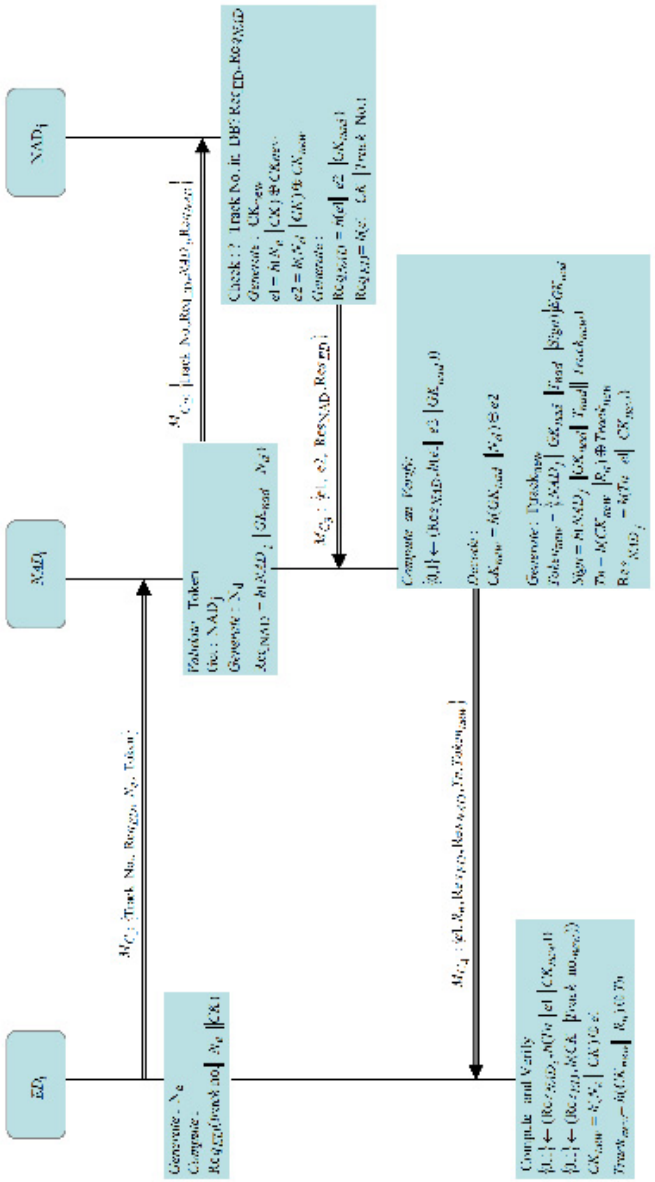


Figure 5.11. Example Authentication when NADs are cooperating

5.6. Performance Analysis

In this section, we provide an analysis of the proposed protocol using the AVISPA evaluation platform.

Note that the tool was previously overviewed. It mainly runs on UNIX(LINUX) but in our case, we virtualized our existing Windows 10 machine for this purpose. Primarily, it is a graphics as well as a push-buttoned tool developed solely for validating security protocols. Its simplicity in use is due to its providing modularised and expressive formal language. It also integrates various automatic protocol analysis techniques. Generally, overall, it is quite robust and at the same time maintains a constant performance as well as scalability.

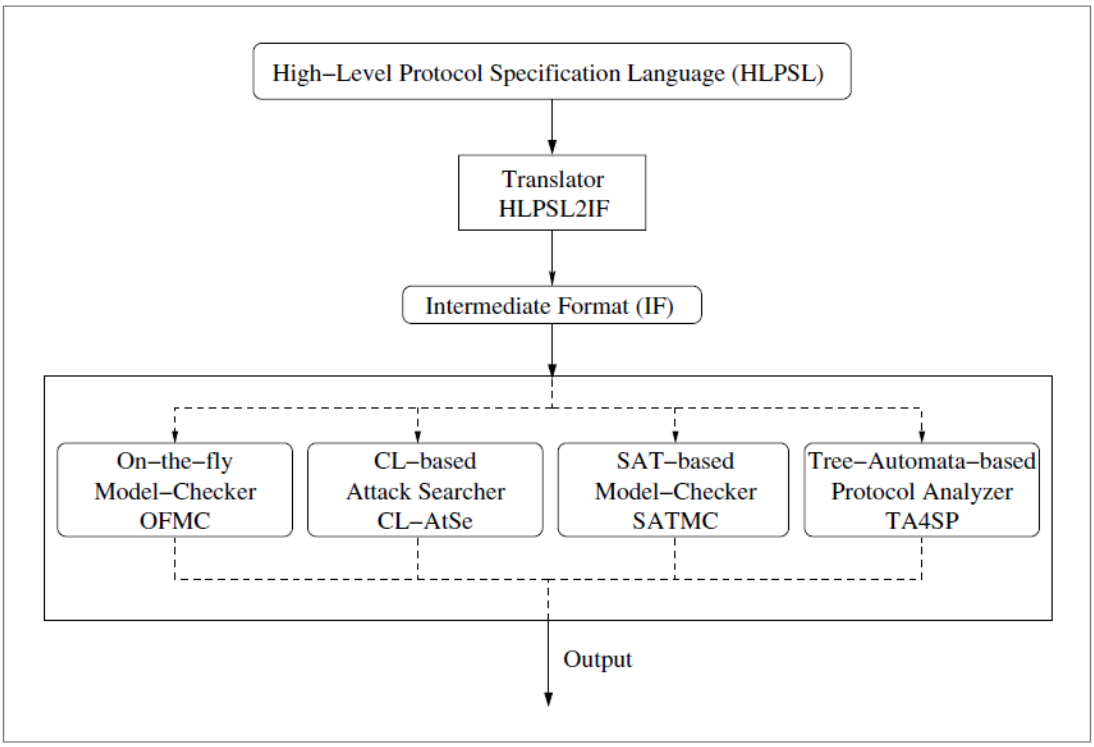


Figure 5.12. The AVISPA Platform

Its architecture is summarised in the figure above. A High-Level Protocol Specification Language (HLPSL) enables users to specify the protocol and its properties. In essence, the HLPSL language is quite modular, expressive and facilitates the specification of control-flow patterns, data structures, different cryptographic operators and their algebraic properties, alternative adversary models, as well as complex security properties. Included is an HLPSL2IF Translator) which transforms a user a user-defined security problem into an equivalent specification written in the rewrite-based formalism Intermediate Format IF.

To test our protocol, we need to fully declare its goals:

```
goal

     authentication_on  group_key

     authentication_on session_key

      secrecy_of sec_m_Key, sec_v_Key

end goal
```

1.6.1 Analysis

In this section, we provide an analysis of the security provided by the proposed scheme as follows:-

Mutual authentication: As mentioned earlier, it is mandatory that all the SG's objects and devices that are D2D communications compliant mutually authenticate with the 3GPP network using the supported AKA authentication framework. Once accomplished connection requests can securely be channeled between the SG device(object) and 3GPP network as it is now guaranteed that all terminals (entities constituting the SG system) are legitimate. Note that the connection request message has the remote SG devices' broadcasted HMAC code. The 3GPP network will use a locally stored HMAC key to verify the legitimacy of the broadcasting entity (device). The system also facilitates peer SG devices to mutually authenticate via an unsecured available channel. To accomplish that, a randomly generated HMAC key is also distributed from the 3GPP network to the devices involved. As this key once-off key is only exchanged via secure channels, attackers are unable to mislead a responding device by replaying previously exchanged messages, neither can a legitimate SG device be impersonated using another set of DHKE messages.

Secure data transmission: In our proposal, we have assumed that data is exchanged only after sessions have been authenticated. As the session keys are generated by ECDH and only transmitted via secured channels privacy-compromising is ruled out Note that ECDH is based on Computational Diffie-Hellman (CDH) problem, adversaries cannot find the symmetric key used. Hence only legitimized parties can read content messages but even an intermediary

like the 3GPP network will not have knowledge of the actual key used. for the particular session.

Session key backward/ forward secrecy: The scheme guarantees that attackers cannot retrieve keys from previous or future sessions. This is to guard against situations where a party has exited the SG and then afterward be used for malicious actions in the SG. Similarly, new entrants may not be able to exploit previous transactions. We note that the session key backward/forward secrecy is consolidated by the fact that the stored HMAC keys are only used for message authentication and verification purposes.

Device anonymity: The scheme uses device pseudo identities. As such attackers may only be aware that live sessions exist on the network, but would not be able to decode that sender and destination's real identities and locations.

Traceability: The scheme requires that a confirmation message be sent upon successful connection establishment. In that way, if too many failed attempts are detected on one particular point, that would serve as an indicator that attackers are attempting to infiltrate.

Message non-repudiation: It is procedural with the proposed scheme that messages be sent via a secured channel, or the insecure channel with either HMAC code or a sequence number appended: the broadcast message, DHKE request message, and DHKE response message are protected using HMAC code. In that way, message non-repudiation is guaranteed and ensured.

5.6.2 Performance Evaluation

We carry out a performance analysis of the proposed scheme, and in this regard, we compare it to similar protocols such as the 5G-IoT D2D [82]. Our focus is on performance aspects such as computational loads, communication overheads, memory requirements for protocol execution, latencies incurred by unknown attacks as well as energy efficiency.

Computational overhead

We use the Bouncy Castle API which is quite similar to the Java Cryptography Architecture (JCA) is relied upon in executing simulation codes for cryptography calculations and time measurement. All the two are embedded in NetSim's IoT library.

The differences in terms of requirements for cryptographical functions used in each scheme, as well as key size are taken into account when running the simulations. However, we ultimately used a 128-bit key throughout to simply In comparing computational overheads we focus on the experimental computational time of running each of the algorithms. Obtained results in Figure 5.13, show that the proposed scheme, by comparison, requires less computational time.

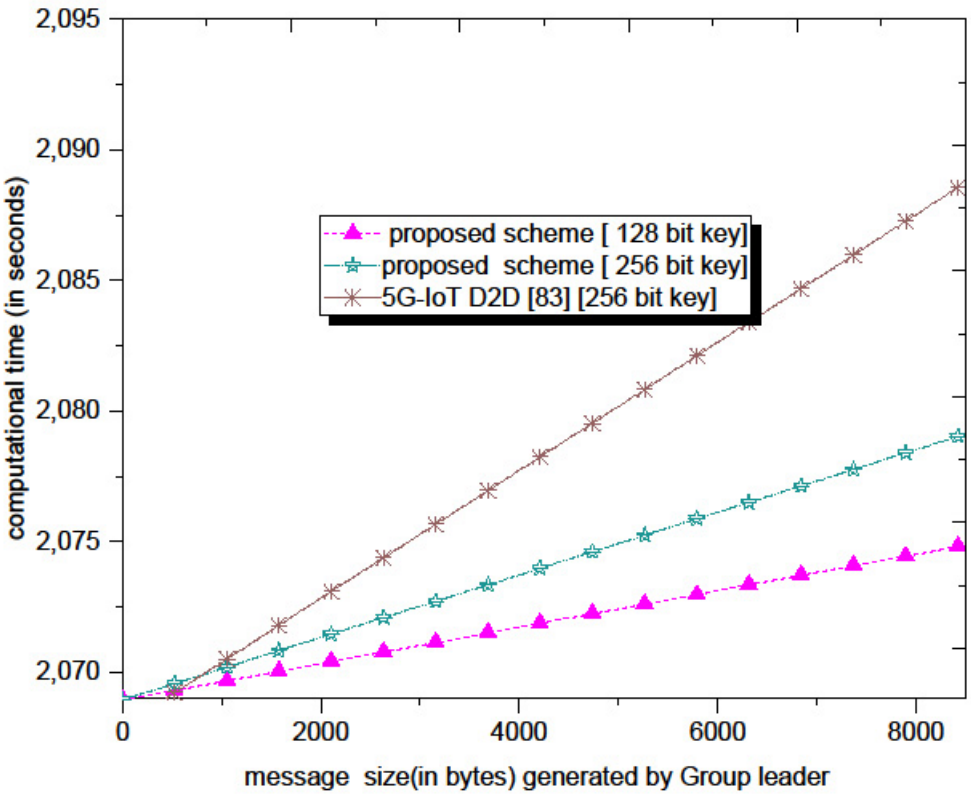


Figure 5.13: Computational time comparisons

Moreover, by further analysis, it is ascertained that the proposed scheme, by comparison, is 27 % faster than the 5G-IoT D2D scheme.

Transmission overhead

One of the scheme's desirable objectives is that of keeping transmission overheads to a minimum since the environment is naturally bandwidth constrained. The overheads for the scheme comprise the aggregate number of signaling messages exchanged, the length of controlling

messages in the protocol, and the supported data rate of the network. We maintain propagation distances of 300 meters and a propagation speed of $3\times10^{8} m/s$.

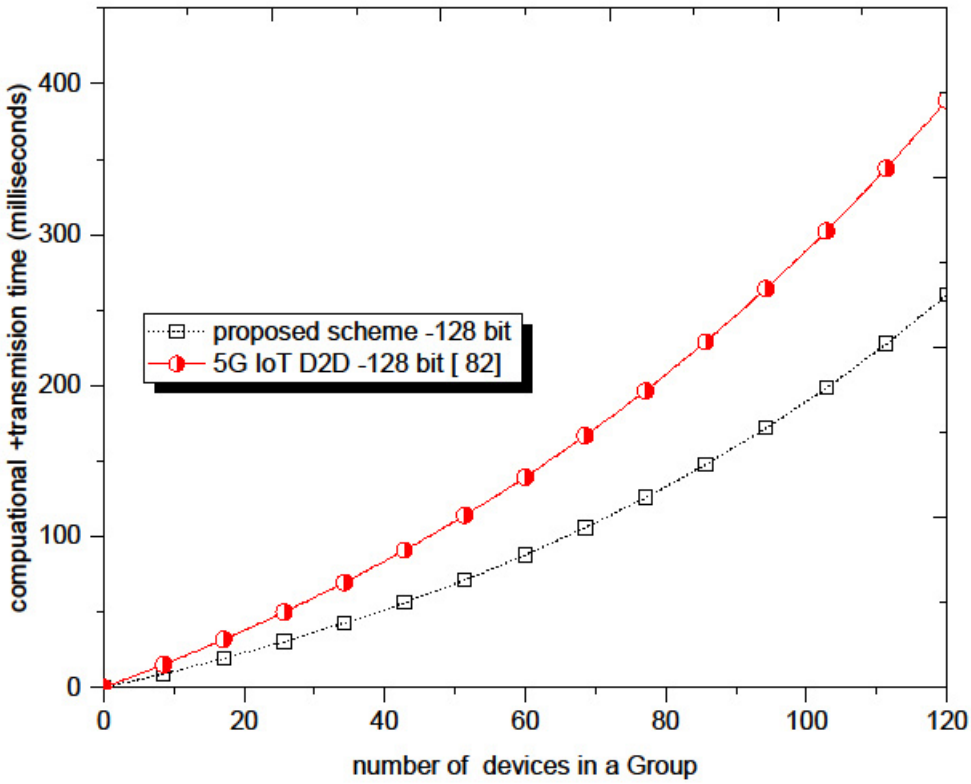


Figure 5.14: Transmission overheads

For the propagation delays, inter-site distance (ISD)/2 250 meters is chosen to simulate the average UE-UE distance, and the propagation speed of the signals is 3 108 ms-1, and uplink and downlink data rates of 30 Mbps and 60 Mbps respectively. All keys have a fixed length of 250 bits, The 5G-GUTI 's length s 100 bits, 5G-SUCI is 256 bits, the message tag is 8 bits, a random nonce is 32 bits, the timestamp is 32 bits, and the session ID is 64 bits. Several simulation runs are carried out and then results are averaged Overall the proposed scheme minimizes transmission overheads as can be observed in the results plotted in Figure 5.14.

Average delay

Since the proposed protocol operates in an uncertain environment in terms of attacks by adversaries, we assume that the protocol will re-initialize each time an attack is registered Thus we measure the average time required for the protocol to accomplish a task under threats of unknown and unanticipated attacks. This part of the simulation is run in MATLAB 2021a.

The in built finddelay function (in MATLAB) uses the xcorr function to determine the cross-correlation between each pair of signals at all possible lags specified by the user. A fragment of the syntax is as follows:

```
r = xcorr(x,y)

r = xcorr(x)

r = xcorr(__,maxlag)

r = xcorr(__,scaleopt)

[r,lags] = xcorr(__)
```

Using the syntax above, a normalized cross-correlation between each pair of signals is then calculated.

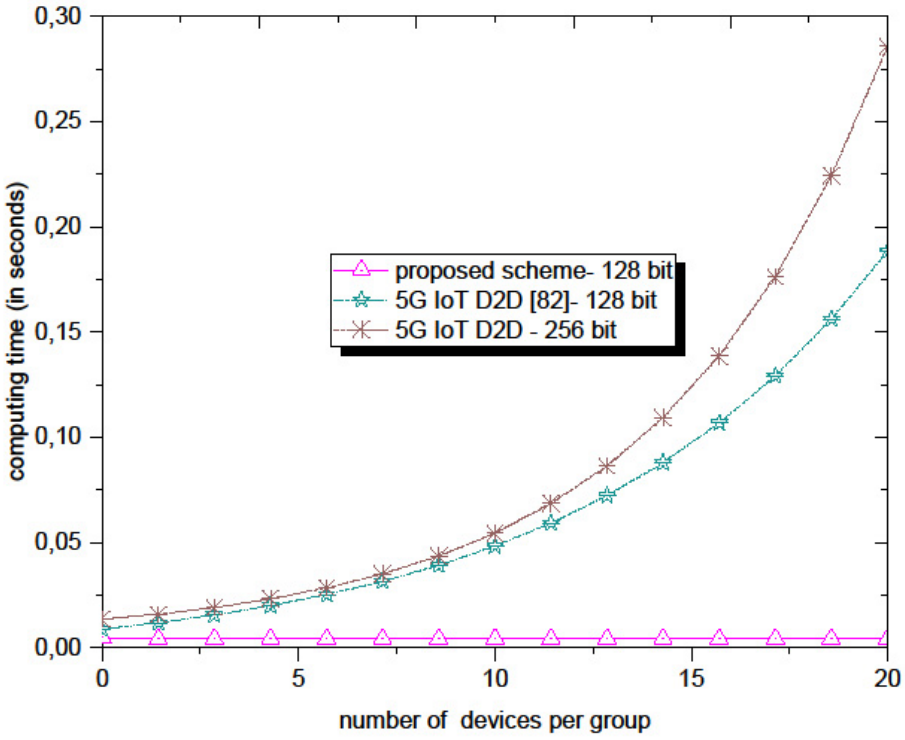


Figure 5.15: Average mean execution delays

The estimated delay is given by the negative of the lag for which the normalized cross-correlation has the largest absolute value. The simulation is run several times and time delays (representing the average time required for the scheme to complete its tasks) are averaged.

Figure 5-15 plots the mean execution times. From the same graph, the proposed scheme registers a lower time delay by comparison.

E. Energy consumption for UE

Energy consumption and efficiency thereof of any security-related protocol will generally be a function of the volumes of signaling messages exchanged due to the cryptographic algorithms used and the times taken to transmit the same signaling messages. Using the LTE data transfer power model carried out in [83], and related approaches in [84], and [85], we compare by estimating the energy consumption of the proposed versus the 5G IoT D2D [82] and LIKE [86].

:

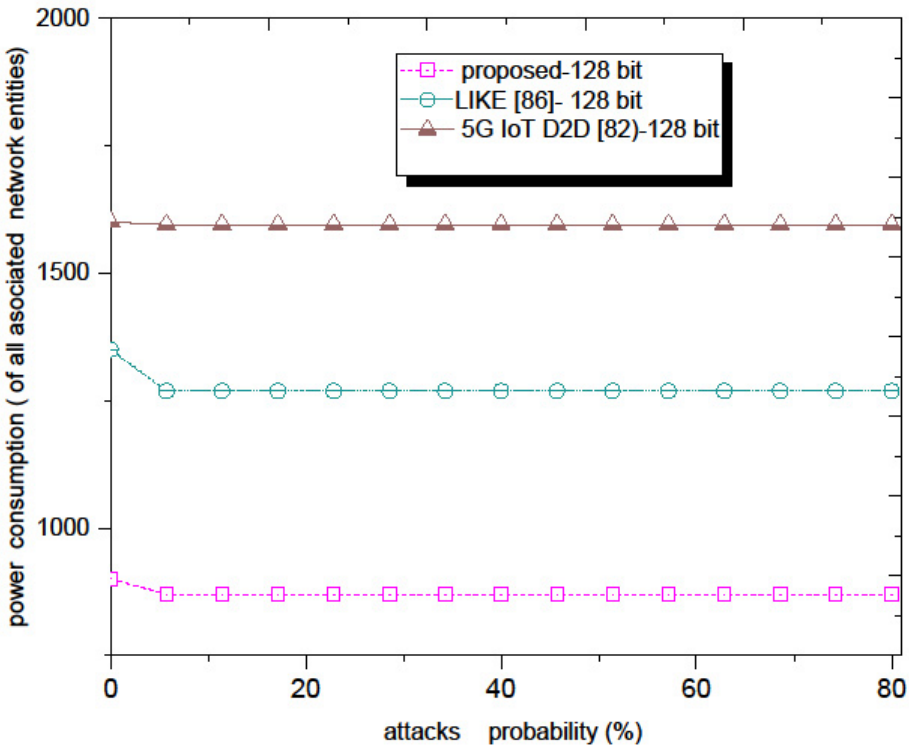


Figure 5.16: Energy efficiency of the various schemes

Using the approaches elaborated in [83], [84], and [85], the average energy consumptions for the three schemes are computed and plotted in Figure 5.16. By comparison, the proposed is much more energy-efficient as it comes much lesser energy for its executions Partly this is

attributed to it relying on the HMAC to replace the power-consuming asymmetric ECDSA and use ECDH to replace the power-consuming modular exponentiation based DHKs. Its use of relatively shorter signaling messages also helps to save energy.

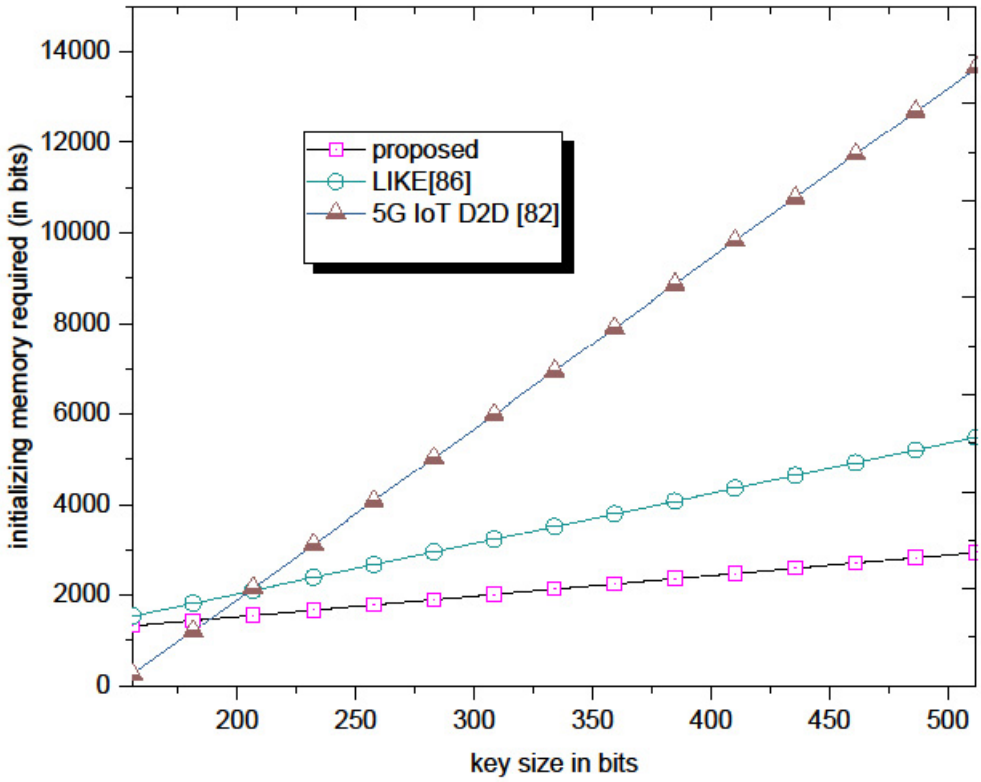


Figure 5.17: Storage overheads versus key size

Memory (storage) overheads

The amount of memory to initialize a protocol is explored here. Because of the resource-constrained nature of most SG devices, objects, and systems, we need to maintain much smaller storage overheads. Example overhead includes items such as key parameters and all distributed materials from TA including pseudo-identities, tokens, and private keys. The computed storage overheads for the proposed and two other schemes namely, 5G IoT D2D [82], and LIKE [86] are plotted in Figure 5.17 in which it can be observed that the proposed scheme has the lowest storage overhead for initialization irrespective of key size.

5.7. Summary Chapter Conclusions

5G wireless networks systems have incorporated D2D communications and its related technologies to leverage many advances such as adequate coverage and network availability. However, both security and privacy threats have also increased due to the number of interconnected entities (devices and objects). Furthermore, the fact that D2D communications facilitate direct interconnectivity among devices in proximity further increases potential security threats. However, the facilitation of direct connectivity between devices in the absence of the core network (due to failure or no coverage) enhances fail-safe connectivity. This is because, among other things, security and privacy schemes that rely on network availability for key authentications will still operate under the circumstances. A congested network may also lead to increased latencies and thus an alternative would be to use the D2D communications channels which may offer better QoS during the congestion period experienced by the core network. Thus in this chapter, we paid attention to the Fog Computing paradigm in concurrency with D2D communications to directly respond to the latency minimization issue as well as network availability. The latter is more important to critical mission services and applications. The paradigm t exploits the fog layer, which is the interfacing layer between the core and peripheral network sections to drastically reduce latencies as well as boost the limited computing powers in resource-constrained devices. It can also provide network context information which ultimately is used by fog applications and services to optimize context-awareness. Its support for location-awareness; means it can fully support device mobility which is a direct booster for location-based services and applications. Fog computing easily provides a local overview whereas a global overview will still be provided by cloud computing.

We proposed a security framework specifically to provide adequate surveillance in an SG network. The associated surveillance cameras in particular targeted areas work in a collaborative manner by forming groups. In that way, the volumes of data exchanges between the various entities are aggregated at the group level before being relayed to the next entity. In the process, the aggregated data traverses several intermediate relaying units. Our proposed framework exploits the fog computing layer to reduce any undesired end-to-end latencies due to constraint computing resources within the devices. Besides, the Fog layer has several entities such as gNBs and other wireless access points hence it is necessary that within this layer level, interaction among the entities is facilitated so that loads can be evened.

Key operational derails of the framework are discussed so is the security analysis. Overall, it is found to suffice in most security and privacy requirements

We also further propose and analyze a lightweight encryption-based authentication protocol for a surveillance service. In so doing our focus is on both the protocol's security resilience and robustness, communication, computational as well as energy efficiencies. Lightweight encryption was chosen since the surveillance camera units themselves are categorized as resource-constrained in terms of both computing power and battery life.

6: Conclusions and Future Research

6.1 Introduction

The gradual evolving of wireless networks has finally resulted in the deployment of the 5G cellular wireless networks which are projected to provide high bandwidth capacities and thus data rates, reduced end-to-end delays for time-sensitive applications and services, and lowered energy consumption by the interconnected devices. D2D communication technology is expected to be widely deployed in the 5G networks to enhance overall network performance. That is, comparatively, D2D communication technology has an added advantage of improving throughput, round trip latencies, and spectrum utilization, in wireless (cellular) networks The technology facilitates direct linkages between device peers, without the involvement of an intermediary as is the case with current communication trends. Essentially, the deployment of D2D communications technology together with the emergency of IoT networking concept has afforded a new paradigm platform network, that reliably facilitates data exchange among devices in proximity. It is geared towards the need the improve network performance where short-range communications are concerned, as well as supporting proximity-based services.

However, the relentless growth in the number of connected users, networked devices, the next-generation IoT-based 5G cellular networks has resulted in the novel as well as would be services and application, most of which are security-sensitive. It is thus of paramount importance that security issues be addressed. A posing challenge is that the devices are mostly resource-constrained in both power and computing. As such, it is not practical to implement present day as well as traditional security frameworks and protocols under such a scenario, unless strides are taken towards the improvements of data throughput rates, higher bandwidth provisioning, lower round trip latencies, enhanced spectral efficiencies, and energy efficiency (leading to even lower power consumption, by the already constrained devices). We thus in this work have focused on robust as well as reliable mutual authentication among communicating peers. In the absence of stringent privacy as well as security guarantees, data exchanged via the D2D communication links can easily be vulnerable to various attacks, e.g., eavesdrops, Man-in-the-Middle (MitM), and impersonation attacks. A user's identity can also be easily compromised.

The focus, therefore, is to secure D2D communication services such that user privacy and identity is never compromised. There is a tendency that a significant number of D2D applica-

tions and services will involve grouped devices and that they will be linked over several domains rather than within a single locality. In this kind of scenario, security issues are addressed taking into consideration that users are located under different domains and that the D2D communication spans over several domains, operated by different operators, and with varying security policies.

The overall work herein centered on the preservation of privacy through robust as well as reliable authentication mechanisms. It was noted that overall, the various IoT systems, services, and applications can only be effective as well as secure assuming the associated D2D communications sessions have been successfully verified and authenticated. Because the majority of would-be D2D services and applications will involve the collaborative interaction of several elements in a group, it may be necessary to advocate for group-based authentication and key agreement (AKA) protocol approaches for achieving effective authentication. These will be expected to satisfy security requirements such as confidentiality, mutual authentication, privacy preservation, integrity and most importantly utilizing a common and single security (encryption) key during the communication sessions in the IoT network. Such protocols need to inherently achieve efficacy in maintaining the group key unlink-ability as well as generate minimal overheads that otherwise may lead to network congestion. Summarily the key security-related challenges in IoT elated communications include but are not limited to:

- User security. e.g. user devices such as smart meters autonomously frequently relay data to the Utility center and as such, there is a possibility that the exchanged data may be intercepted by intruders to figure out devices being used in the home, infer consumer's activities, as well as times when the home is vacated. The multitudes of intelligent IoT network pillar devices may eventually be used as attack entry points. Furthermore, the vast areas of coverage by the smart grid itself make network monitoring and management extremely difficult.
- Physical security: Most components that build the various IoT systems, applications, and services are placed remotely (i.e. outside buildings). As such the fact that there are many insecure physical locations makes Smart Grids prone and vulnerable to physical access.

The lifetime span disparities between power systems components versus those of IoT-driven technologies and infrastructures implies that the two exist alongside the older equipment act-

ing as a weak entry point for malicious attacks. The old equipment may also be incompatible with the current power system devices hence compromising overall delivery efficiencies.

Implicit trust relationships among traditional power devices make D2D communications control systems are vulnerable to data spoofing. This is because the condition of one device may adversely affect the actions of another.

Disparities in Team's background may lead to considerable vulnerabilities. This is attributed to frequent uncoordinated communication between teams which leads to lots of bad decisions as well.

The utilizing of IP-compatible equipment in the IoT infrastructure makes it inherently vulnerable to traditional IP-based network attacks such as IP spoofing, Tear Drop, Denial of Service, and others. The conglomerating of several stakeholders (investors) in the IoT network, might give raise insider attacks."

Overall, in the core work, mainly an overview of D2D communication, as well as related privacy issues, were explored. The work also discussed the requirements for the fulfillment of privacy for D2D communication-based services and applications. An analysis of selected privacy authentication protocols for D2D communication is carried out. The work also proposes a Fog Computing paradigm-based Privacy/security framework for D2D based applications and services as it is seen to have the potential to drastically reduce latencies experienced with Cloud computing. The framework exploits the fog layer, which is the interfacing layer between the core and peripheral network sections to drastically reduce latencies as well as boost the limited computing powers in resource-constrained devices. It can also provide network context information which ultimately is used by fog applications and services to optimize context-awareness. Its support for location-awareness; means it can fully support device mobility which is a direct booster for location-based services and applications. Fog computing easily provides a local overview whereas a global overview will still be provided by cloud computing.

6.2. Future Research Directions

Overall, as far as privacy for D2D communications-based applications and services is concerned, in the future, the work will pillar on focusing on secure and lightweight mutual authentication schemes that provide full reliance to malicious attacks, as well as security robust-

ness and energy efficiency. It is also important that such schemes should operate across multiple domain networks i.e. interoperability must be ensured. The research will also pay attention to the potent concept of the fog computing paradigm as it drastically reduces latencies that otherwise characterize cloud computing.

Whereas the Fog Computing paradigm places its key servers at network peripherals so as to reduce latencies, however, it is still an unresolved issue as to how afar energy-effective cooperation policies can be implemented among the deployed servers to enhance end users' quality of experience (QoE). In the future, we will thus also explore the possibility of designing a cooperative fog computing framework that will optimally redistribute workload among all cooperating servers. In all cases special focus will be on energy efficiency [74], [75], [76], [77], [78] as well as ensuring total privacy in various spheres of industry and general life applications and services [79], [80], [81].

References

[1] C. Lee and A. Fumagalli, "Internet of Things Security - Multilayered Method For End to End Data Communications Over Cellular Networks," 2019 IEEE 5th World Forum on Internet of Things (WF-IoT), Limerick, Ireland, 2019, pp. 24-28, doi: 10.1109/WF-IoT.2019.8767227.

[2] M. Haus, M. Waqas, A. Y. Ding, Y. Li, S. Tarkoma and J. Ott, "Security and Privacy in Device-to-Device (D2D) Communication: A Review," in IEEE Communications Surveys & Tutorials, vol. 19, no. 2, pp. 1054-1079, Secondquarter 2017, doi: 10.1109/COMST.2017.2649687.

[3] A. V. Bastos, C. M. Silva and D. C. da Silva, "Assisted routing algorithm for D2D communication in 5G wireless networks," 2018 Wireless Days (WD), Dubai, 2018, pp. 28-30, doi: 10.1109/WD.2018.8361688.

[4] M. Abdollahi, M. Abolhasan, N. Shariati, J. Lipman, A. Jamalipour and W. Ni, "A Routing Protocol for SDN-based Multi-hop D2D Communications," 2019 16th IEEE Annual Consumer Communications & Networking Conference (CCNC), Las Vegas, NV, USA, 2019, pp. 1-4, doi: 10.1109/CCNC.2019.8651752.

[5] M. Wang, Z. Yan, B. Song and M. Atiquzzaman, "AAKA-D2D: Anonymous Authentication and Key Agreement Protocol in D2D Communications," 2019 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI), Leicester, United Kingdom, 2019, pp. 1356-1362, doi: 10.1109/SmartWorld-UIC-ATC-SCALCOM-IOP-SCI.2019.00248.

[6] P. Fremantle, B. Azi, J. Kopeck and P.Scot, "Federated Identity and Access Management for the Internet of Things", SIOTP Proceedings, September 2014.

[7] M. R. Palattella, E, Nicola A. Xa.r Vilajosana, T. Watteyne, L.A. Grieco, I G. Boggia, and I. Dohler," Standardized Protocol Stack for the Internet of Things", IEEE Communications Surveys & Tutorials, Volume 15, Number 3. Quarter, 2013.

[8] A. Messias , C. Souza, and R.A.Amazonas, "A New Internet of Things Architecture with Cross-Layer Communication", Proceedings of the Seventh International Conference on Emerging Networks and Systems Intelligence, EMERGING 2015.

[9] T. Markmann, T.C. Schmidt and M. Wählisch, "Federated End-to-End Authentication for the Constrained Internet of Things Using IBC and ECC", SIGCOMM '15 August 17-21, 2015, London, United Kingdom.

[10] R.Bonetto, N. Bui, V.Lakkundi, A. Olivereau, A. Serbanati and M. Ross, "Secure Communication for Smart IoT Objects: Protocol Stacks, Use Cases and Practical Examples.ITU. ITU-T Y.2060 . "Intro to Internet of Things" https://www.itu.int/rec/T-REC-Y.2060-201206-I/en, June, 2012.

[11] IEEE 802.15.4e-2012: IEEE Standard for Local and Metropolitan Area Networks - Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs) Amendment 1: MAC Sublayer, Institute of Electrical and Electronics Engineers Std., 16 April 2012.

[12] Piro, G. Boggia and L. A. Grieco, "A standard-compliant security framework for Low-power and Lossy Networks, draft-piro-6tisch-security-issues-01 (work in progress)", IETF 6TiSCH WG, December 14, 2013.

[13] Syed Muhammad Sajjad_, Muhammad Yousaf.Security Analysis of IEEE 802.15.4 MAC in the context of Internet of Things (IoT). 2014 IEEE Conference on Information Assurance and Cyber Security (CIACS).

[14] B. Aboba, L. Blunk, J. Vollbrecht, J. Carlson, and H. Levkowetz, "RFC 3748: Extensible Authentication Protocol (EAP)," IETF Request For Comments, ttp://www.ietf.org/ rfc/rfc3748.txt, Jun. 2004.

[15] D. Forsberg, Y. Ohba, B. Patil, H. Tschofenig, and A. Yegin, "RFC5191: Protocol for Carrying Authentication for Network Access (PANA)," IETF Request For Comments, http://tools. ietf.org/rfc/rfc5191.txt, May 2008.

[16] MUlti-cloud Secure Applications (MUSA) Project. Call: H2020-ICT-2014-1: http://www.musa-project.eu.

[17] Cloud-of-Things - (ClouT) Project. Call: FP7-ICT-2013- EU-Japan. http://clout-project.eu.In-network programmability for next-generation personal cloud service sup-port(INPUT) Project. Call: H2020-ICT-2014-1,:http://input-project.eu.

[18] J. Cao *et al*., "A Survey on Security Aspects for 3GPP 5G Networks," in *IEEE Communications Surveys & Tutorials*, vol. 22, no. 1, pp. 170-195, Firstquarter 2020, doi: 10.1109/COMST.2019.2951818.

[19] M. Gomba and B. Nleya, "Architecture and security considerations for Internet of Things," 2017 Global Wireless Summit (GWS), Cape Town, 2017, pp. 252-256, doi: 10.1109/GWS.2017.8300477.

[20 L. P. Bopape, B. Nleya and P. Khumalo, "A Privacy and Security Preservation Framework for D2D Communication Based Smart Grid Services," 2020 Conference on Information Communications Technology and Society (ICTAS), Durban, South Africa, 2020, pp. 1-6, doi: 10.1109/ICTAS47918.2020.233995.

[21] M. Gomba and B. Nleya, "Overview Access and Control Considerations for Internet of Things," 2018 International Conference on Advances in Big Data, Computing and Data Communication Systems (icABCD), Durban, 2018, pp. 1-7, doi: 10.1109/ICABCD.2018.8465435.

[22] K. Loupos et al., "Cognition Enabled IoT Platform for Industrial IoT Safety, Security and Privacy — The CHARIOT Project," 2019 IEEE 24th International Workshop on Computer-Aided Modeling and Design of Communication Links and Networks (CAMAD), Limassol, Cyprus, 2019, pp. 1-4, doi: 10.1109/CAMAD.2019.8858488.

[23] P. K. Singh, R. Saxena, U. Dubey, A. Raj, B. M. Sahoo and V. Bibhu, "Smart Security System Using IOT," 2020 International Conference on Intelligent Engineering and Management (ICIEM), London, United Kingdom, 2020, pp. 392-395, doi: 10.1109/ICIEM48762.2020.9160144.

[24] S. Liu, K. Yue, Y. Zhang, H. Yang, L. Liu and X. Duan, "The Research on IoT Security Architecture and Its Key Technologies," 2018 IEEE 3rd Advanced Information Technology, Electronic and Automa-

tion Control Conference (IAEAC), Chongqing, 2018, pp. 1277-1280, doi: 10.1109/IAEAC.2018.8577778.

[25] K. Lounis and M. Zulkernine, "Attacks and Defenses in Short-Range Wireless Technologies for IoT," in IEEE Access, vol. 8, pp. 88892-88932, 2020, doi: 10.1109/ACCESS.2020.2993553.

[26] M. Gomba and Bakhe Nleya, "Architecture and security considerations for Internet of Things," 2017 Global Wireless Summit (GWS), Cape Town, 2017, pp. 252-256, doi: 10.1109/GWS.2017.8300477.

[27] L. P. Bopape, B. Nleya and P. Khumalo, "A Privacy and Security Preservation Framework for D2D Communication Based Smart Grid Services," 2020 Conference on Information Communications Technology and Society (ICTAS), Durban, South Africa, 2020, pp. 1-6, doi: 10.1109/ICTAS47918.2020.233995.

[28] A. Bechtsoudis and N. Sklavos, "Side-Channel Attacks Cryptanalysis against Block Ciphers Based on FPGA Devices," 2010 IEEE Computer Society Annual Symposium on VLSI, Lixouri, Kefalonia, 2010, pp. 460-461, doi: 10.1109/ISVLSI.2010.104.

[29] L. Melki, S. Najeh and H. Besbes, "System performance of two-way decode-and-forward relaying assisted D2D communication underlaying cellular networks," 2016 International Symposium on Signal, Image, Video and Communications (ISIVC), Tunis, 2016, pp. 270-275, doi: 10.1109/ISIVC.2016.7893999.

[30] S. M. M. Islam, F. A. Qazi, M. F. Iskander, Z. Yun and G. Sasaki, "Advanced directional networking: LTE vs WiFi radios," 2017 IEEE International Symposium on Antennas and Propagation & USNC/URSI National Radio Science Meeting, San Diego, CA, 2017, pp. 189-190, doi: 10.1109/APUSNCURSINRSM.2017.8072137.

[31] Y. Zheng and X. Zhang, "Path sensitive static analysis of web applications for remote code execution vulnerability detection," 2013 35th International Conference on Software Engineering (ICSE), San Francisco, CA, 2013, pp. 652-661, doi: 10.1109/ICSE.2013.6606611.

[32] O. Hayat, R. Ngah, S. Z. Mohd Hashim, M. H. Dahri, R. Firsandaya Malik and Y. Rahayu, "Device Discovery in D2D Communication: A Survey," in IEEE Access, vol. 7, pp. 131114-131134, 2019, doi: 10.1109/ACCESS.2019.2941138.

[33] P. Mach, Z. Becvar and T. Vanek, "In-Band Device-to-Device Communication in OFDMA Cellular Networks: A Survey and Challenges," in IEEE Communications Surveys & Tutorials, vol. 17, no. 4, pp. 1885-1922, Fourth quarter 2015, doi: 10.1109/COMST.2015.2447036.

[34] Y. Cai, Y. Ni, J. Zhang, S. Zhao and H. Zhu, "Energy efficiency and spectrum efficiency in underlay device-to-device communications-enabled cellular networks," in China Communications, vol. 16, no. 4, pp. 16-34, April 2019, doi: 10.12676/j.cc.2019.04.002.

[35] W. Shen, W. Hong, X. Cao, B. Yin, D. M. Shila and Y. Cheng, "Secure key establishment for Device-to-Device communications," 2014 IEEE Global Communications Conference, Austin, TX, 2014, pp. 336-340, doi: 10.1109/GLOCOM.2014.7036830.

[36] Jiang, Q., Khan, K. M., Lu, X., Ma, J. and He, D., “A privacy-preserving three-factor authentication protocol for e-health clouds.”, *The Journal of Supercomputing*, v. 72, n. 10, pp. 3826–3849, 2016.

[37] Y. Sun, Q. Wen, H. Sun, W. Li, Z. Jin, H and Zhang,”An Authenticated Group Key Transfer Protocol Based on Secret Sharing”, Procedia Engineering, Volume 29, 2012, Pages 403-408,

[38] Shen, J., Gui, Z., Ji, S., Shen, J., Tan, H. and Tang, Y., “Cloud-aided lightweight certificateless authentication protocol with anonymity for wireless body area networks.”, *Journal of Network and Computer Applications*, v. 106, pp. 117-123, 2018.

[39] Tzvetalin S. Vassilev, Andrew Twizell, "Cryptography: A Comparison of Public Key Systems", *Algorithms Research*, Vol. 1 No. 5, 2012, pp. 31-42. doi: 10.5923/j.algorithms.20120105.01.

[40] Mohit, P., Amin, R., Karati, A., Biswas, G. P., Khan, M. K., "A standard mutual authentication protocol for cloud computing based health care system.", *Journal of medical systems*, v.41, n. 4, pp. 50, 2017.

[41] Thomas Genet. A Short SPAN+AVISPA Tutorial. [Research Report] IRISA. 2015. ffhal-01213074v1 https://hal.inria.fr/hal-01213074v1

[42] U. N. Kar and D. K. Sanyal,”An overview of device-to-device communication in cellular networks”, ICT Express, Volume 4, Issue 4,2018, Pages 203-208,

[43] https://www.prnewswire.com/news-releases/prose-proximity-services-for-lte--5g-networks-2017-2030---opportunities-challenges-strategies--forecasts-300396915.html

[44] N. Kahani, K. Elgazzar and J. R. Cordy, "Authentication and Access Control n e-Health Systems in the Cloud," 2016 IEEE 2nd International Conference on Big Data Security on Cloud (BigDataSecurity), IEEE International Conference on High Performance and Smart Computing (HPSC), and IEEE International Conference on Intelligent Data and Security (IDS), New York, NY, 2016, pp. 13-23, doi: 10.1109/BigDataSecurity-HPSC-IDS.2016.43.

[45] S. S. Sahoo and S. Mohanty, "Cloud-Assisted Privacy-Preserving Authentication Scheme for Telecare Medical Information Systems," *2018 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)*, Indore, India, 2018, pp. 1-6, doi: 10.1109/ANTS.2018.8710128

[46] A. Paula G. Lopes and Paulo R. L. Gondim , “Mutual Authentication Protocol for D2D Communications in a Cloud-Based E-Health System”, Sensors, April 2020, doi:10.3390/s20072072

[47] Zhang, A., Wang, L., Ye, X., & Lin, X. (2017). “Light-weight and robust security-aware d2d-assist data transmission protocol for mobile-health systems”, *IEEE Transactions on Information Forensics and Security*, Vol.12, n.3, pp.662-675.

[48] R. H., Hsu, J., Lee, T. Q., Quek, and J. C., Chen, “GRAAD: Group Anonymous and Accountable D2D Communication in Mobile Networks.”, *IEEE Transactions on Information Forensics and Security*, vol.13, no.2, pp.449-464, 2018.

[49] Chiou, S., Ying, Z. and Liu, J., "Improvement of a privacy authentication scheme based on cloud for medical environment”, *Journal of medical systems*, v. 40, n. 4, pp.101, 2016.

[50] M, Wang, and Z., Yan, “Privacy-preserving authentication and key agreement protocols for D2D group communications”, *IEEE Transactions on Industrial Informatics*, vol.14, no.8, pp.3637-3647, 2018.

[51] T. Toukabri, G. Steve , T. Kwong and U. H.Afifi, “Hybrid Model for LTE Network-Assisted D2D Communications “Part of the Lecture Notes in Computer Science book series (LNCS, volume 8487)

[52] Stallings, W. Cryptography and Network Security: Principles and Practice; Pearson: Upper Saddle River, NJ,USA, 2017.

[53] B. Seok , J. Costa , S. Sicato, T. Erzhena , C. Xuan ,Yi Pan and J. H. Park, “Secure D2D Communication for 5G IoT Network Based on Lightweight Cryptography” Applied Sciences, December, 2019.

[54] Rezvani, B.; Diehl, W. Hardware Implementations of NIST Lightweight Cryptographic Candidates: A First Look. In Proceedings of the NIST Lightweight Cryptography Workshop 2019, Gaithersburg, MD, USA, 4–6 November 2019

[55] J. Yao, T. Wang, M. Chen, L. Wang, and G. Chen, ``GBS-AKA: Group based secure authentication and key agreement for M2M in 4G network," in Proc. IEEE Int. Conf. Cloud Computing. Research Innovation. (ICCCRI), May 2016,pp. 42-48.

[56] G. Singh, D. D. Shrimankar," Dynamic Group-Based Efficient Access Authentication, and Key Agreement Protocol for MTC in LTE-A", Wireless Networks, Personal Communications, Published online, 13 April, 2018.

[56] http://www.irisa.fr/lande/genet/span

[57] J. Yao, T. Wang, M. Chen, L. Wang, and G. Chen, ``GBS-AKA: Group-based secure authentication and key agreement for M2M in 4G network," in Proc. IEEE Int. Conf. Cloud Computing. Research Innovation. (ICCCRI), May 2016,pp. 42-48.

[58] Y.-W. Chen, J.-T. Wang, K.-H. Chi, and C.-C. Tseng, ``Group-based authentication and key agreement,' Wireless Personal. Communications, vol. 62, no. 4,pp. 965-979, 2012.

[59] S. M. R. Islam, D. Kwak, M. Humaun and K.-S. Kwak, ‘‘The Internet of Things for health care: A comprehensive survey,’’ IEEE Access, vol. 3, pp. 678–708, Jun. 2015.

[60] K.-R. Jung, A. Park, and S. Lee, ‘‘Machine-type-communication (MTC) Device grouping algorithm for congestion avoidance of MTC oriented LTE network,’’ in Security-Enriched Urban Computing and Smart Grid. Berlin, Germany: Springer, 2010, pp. 167–178.

[61] Y.-W. Chen, J.-T. Wang, K.-H. Chi, and C.-C. Tseng, ‘‘Group-based authentication and key agreement,’’ Wireless Personal. Communications., vol. 62, no. 4, pp. 965–979, 2012.

[62] C. Lai, H. Li, R. Lu, and X. S. Shen, ‘‘SE-AKA: A secure and efficient group authentication and key agreement protocol for LTE networks,’’ Computer. Networks., vol. 57, no. 17, pp. 3492–3510, 2013.

[63] R. Jiang, C. Lai, J. Luo, X. Wang, and H. Wang, ‘‘EAP-based group authentication and key agreement protocol for machine-type communications,’’ Int. Journal of.Distributed Sensor Networks., vol. 9, no. 11, p. 304601, 2013.

[64] D. Choi, H.-K. Choi, and S.-Y. Lee, ‘‘A group-based security protocol for machine-type communications in LTE-advanced,’’ Wireless Networks., vol. 21, no. 2, pp. 405–419, 2015.

[65] A. Fu, J. Song, S. Li, G. Zhang, and Y. Zhang, ‘‘A privacy-preserving group authentication protocol for machine-type communication in LTE/LTE-A networks,’’ Security Communication. Networks., vol. 9, no. 13, pp. 2002–2014, 2016.

[66] L. Wang, H. An and Z. Chang, "Security Enhancement on a Lightweight Authentication Scheme With Anonymity Fog Computing Architecture," in IEEE Access, vol. 8, pp. 97267-97278, 2020, doi: 10.1109/ACCESS.2020.2996264.

[67] Gope, P. orcid.org/0000-0003-2786-0273 (2019) LAAP: Lightweight anonymous authentication protocol for D2D-Aided fog computing paradigm. Computers & Security, 86,pp. 223-237. https://doi.org/10.1016/j.cose.2019.06.003

[68] A. Shafiq, M. Faizan Ayub,K. Mahmood , M. Sadiq,S. Kumari, and C.-Ming Chen, “ An Identity-Based Anonymous Three-Party Authenticated Protocol for IoT Infrastructure”, Hindawi Journal of Sensors, Volume 2020, Article ID 8829319, 17 pages

[69] C. Lai, R. Lu, D. Zheng, H. Li, and X. S. Shen, ‘‘GLARM: Group-based lightweight authentication scheme for resource-constrained machine to machine communications,’’ Computer. Networks., vol. 99, pp. 66–81, Apr. 2016.

[70] P. Khumalo, B.Nleya, A. Gomba, A Mutsvangwa. "Services and Applications Security in IoT Enabled Networks", International Conference on Intelligent and Innovative Computing Applications (ICONIC),2018.

[71] P. Roychoudhury, B. Roychoudhuryand D. K. Saikia, “Hierarchical Group Based Mutual Authentication and Key Agreement for Machine Type Communication in LTE and Future 5G Networks, Security and Communication Networks”, Volume 2017, Article ID 1701243, 21 pages

[72] L. Bopape, B Nleya, P. Khumalo, and A. Mutsvangwa, “A Group Authentication And Data Security Scheme For Smart Metering In Smart Grids”, Ponte International Journal of Research and Sciences, Vol. 75, | No. 11/1 | January 2020 DOI: 10.21506/j.ponte.*2020.1.4, ISSN: 0032-423X.*

[73]. P. Bopape, B. Nleya, and P. Khumalo, "A Privacy and Security Preservation Framework for D2D Communication Based Smart Grid Services," 2020 Conference on Information Communications Technology and Society (ICTAS), Durban, South Africa, 2020, pp. 1-6, doi: 10.1109/ICTAS47918.2020.233995.

[74] B. Nleya and C. Mulangu, "An Overview of GREEN Networking and Power Savings in Optical Backbone Networks," 2018 International Conference on Advances in Big Data, Computing and Data Communication Systems (icABCD), Durban, South Africa, 2018, pp. 1-6, doi: 10.1109/ICABCD.2018.8465402.

[75] B. Nleya and R. Chidzonga, "Overview of power aware — RWA in optical backbone supported networks," 2017 IEEE AFRICON, Cape Town, South Africa, 2017, pp. 446-449, doi: 10.1109/AFRCON.2017.8095523.

[76] P. Khumalo and B. Nleya, "Sleep-Mode/Traffic Grooming Versus Device Reliability Overview," 2018 International Conference on Advances in Big Data, Computing and Data Communication Systems (icABCD), Durban, South Africa, 2018, pp. 1-5, doi: 10.1109/ICABCD.2018.8465416.

[77] A. Mutsvangwa, B. Nleya and M. Dewa, "Optimized Energy-Aware Lightpath Routing Strategy for Translucent Optical Networks," 2019 International Conference on Advances in Big Data, Computing and Data Communication Systems (icABCD), Winterton, South Africa, 2019, pp. 1-6, doi: 10.1109/ICABCD.2019.8851012.

[78] M. Molefe, B. Nleya, R. Chidzonga, L. Bopape and K. Sibiya, "An Energy-Efficient Impairment-Aware Routing Algorithm For Optical Transport Networks," 2021 Conference on Information Communications Technology and Society (ICTAS), Durban, South Africa, 2021, pp. 11-15, doi: 10.1109/ICTAS50802.2021.9395021.

[79] M. Gomba and B. Nleya, "Architecture and security considerations for Internet of Things," 2017 Global Wireless Summit (GWS), Cape Town, 2017, pp. 252-256, doi: 10.1109/GWS.2017.8300477.

[80] M. Gomba and B. Nleya, "Overview Access and Control Considerations for Internet of Things," 2018 International Conference on Advances in Big Data, Computing and Data Communication Systems (icABCD), Durban, South Africa, 2018, pp. 1-7, doi: 10.1109/ICABCD.2018.8465435.pp. 1-7, doi: 10.1109/ICONIC.2018.8601298.

[81] P. Khumalo, B. Nleya, A. Gomba and A. Mutsvangwa, "Services and Applications Security in IoT Enabled Networks," 2018 International Conference on Intelligent and Innovative Computing Applications (ICONIC), Mon Tresor, Mauritius, 2018, pp. 1-7, doi: 10.1109/ICONIC.2018.8601298.

[82] Baskaran SBM, Raja G. A Lightweight Incognito Key Exchange Mechanism for LTE-A Assisted D2D Communication. In: 2017 9th Int. Conf. Adv. Comput. ICoAC 2017; 2018. p. 301–7.

[83] A Close Examination of Performance and Power Characteristics of 4G LTE Networks. ww.cs.columbia.edu/~lierranli/coms6998-7Spring2014/papers/rrclte_mobisys2012.pdf.

[84] C. Guo, Y. Yang, Y. Zhou, K. Zhang and S. Ci, "A Quantitative Study of Energy Consumption for Embedded Security," 2021 IEEE Wireless Communications and Networking Conference (WCNC), 2021, pp. 1-5, doi: 10.1109/WCNC49053.2021.9417382.

[95] N. R. Potlapally, S. Ravi, A. Raghunathan and N. K. Jha, "A study of the energy consumption characteristics of cryptographic algorithms and security protocols," in IEEE Transactions on Mobile Computing, vol. 5, no. 2, pp. 128-143, Feb. 2006, doi: 10.1109/TMC.2006.16.

[86] M. M. Fouda, Z. M. Fadlullah, N. Kato, R. Lu and X. S. Shen, "A Lightweight Message Authentication Scheme for Smart Grid Communications," in IEEE Transactions on Smart Grid, vol. 2, no. 4, pp. 675-685, Dec. 2011, doi: 10.1109/TSG.2011.2160661.