Adaptive Secure Network Model for Dynamic Wireless Mesh Network

By

Ashish Nanda

A dissertation submitted to

Faculty of Engineering and Information Technology

University of Technology Sydney

In fulfilment of the requirements for the award of

Doctor of Philosophy – Computer Systems

June 2018

Dedicated To

My Loving Parents

My Artistic Brother

My Supportive Family

My Motivating Teachers and Mentors

My Crazy Friends

And To Everyone That Reads This

Certificate of Original Authorship

I, Ashish Nanda declare that this thesis is submitted in fulfilment of the requirements for the award of Doctor of Philosophy, in the Faculty of Engineering and Technology at the University of Technology Sydney.

This thesis is wholly my own work unless otherwise reference or acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis.

This document has not been submitted for qualifications at any other academic institution.

Signature of Student:

Production Note:
Signature removed prior to publication.

Date: 2/07/2018

Acknowledgement

I would like to convey my deepest gratitude to my principal coordinating supervisor Dr. Priyadarsi Nanda and my honest appreciation to my co-supervisor Prof. Xiangjian He, external supervisor Dr. Aruna Jamdagni from Western Sydney University and mentor Dr. Meenakshi Mahajan from National Informatics Center, India for their experienced supervision and continuous encouragement throughout my PhD study. I would also like to thank the University of Technology Sydney (UTS), Faculty of Engineering and IT (FEIT), staff members, research assistants, previous and current colleagues at UTS for all the help throughout my doctoral program.

To my friends, I thank you for listening, offering me advice, and supporting me through this journey. Special thanks to Dr. Deepak Puthal, Dr. Karthick Thiyagarajan, Vijaya Chemalamarri, Pulkit Rohilla, Jaime Garcia, Ashish Kapoor, Neha Sharma, Madhumita Takalkar, Abhijeet Takalkar, Alina Rakhi, Manisha Pratihast, Ashish Kumar, Chau Nguyen, Tham Nguyen, Sara Farahmandian, Gaurvi Goyal, Umesh Gupta, Aayushi Singh, Abhilash Kalotra, Akanksha Rana, Divya Krishna, Manushree Sharma, Yukti Mangal, Deepa Singh, Harshitha Mysore, Sidhant Sehegal, Saurabh Sachdeva, Shweta Sachdeva.

I am deeply grateful to my parents Arun Kumar Nanda and Meenakshi Nanda, my brother Ayush Nanda, my grandparents Late Shri Jagdish Prasad Nanda, Susheela Nanda, Late Shri Ram Prakash Sharma, Chanchal Sharma and the rest of my family for their support, constant encouragement and guidance. Most importantly, I would like to express my sincere gratitude to Almighty God for guiding me along this path.

Abstract

We as an advanced civilization rely on communication networks for a lot of important tasks. They are used to share information between vital systems, provide us with our pin-point location, access various digital resources and to stay connected with each other. Due to its necessity and enormity, maintaining and securing such a communication medium is an important task. As most communication networks rely on centralized systems, they are bound by the control of a central entity and are unable to keep up with the current growth of the network and advancements in electronic devices. The next step in an inter-connected world requires a decentralized distributed system that can also provide high levels of security. One possible solution is a dynamic distributed wireless mesh network as it provides all the features of a traditional network along with the flexibility of wireless communication and an infrastructure less distributed setup. The network can be created by connecting mobile or stationary devices together using wireless communication devices (such as smartphones, laptops, hot-spots, etc). As the network is created by multiple devices, it would not break-down if some of the devices were disabled. On the contrary, as the network uses hopping for message transmission using dynamic routes, it can self-heal by creating alternate routes if a device was to fail. As the workings and features of a dynamic mesh network differ from the traditional network, it also requires a modified security framework that can provide high levels of security whilst taking benefit of the dynamic mesh network's unique features.

This thesis investigates the problems and limitations linked to secure dynamic wireless mesh networks and how they can be improved upon. In addition to the routing protocols used and how they can be improved upon, the thesis also elaborates on the various security concerns with such networks. As distributed networks aren't dependent on a central entity, enabling various security features such as authentication are a major challenge. In addition to the decentralized nature of the networks, a single security scheme would not be able to cover the various types of requirements a given scenario in the network might have. Along with authentication, providing end-to-end encryption is also an important component towards ensuring the data travelling through the network is secure and not tampered with. Encryption is also essential in a dynamic wireless mesh network as the data transmitted travels through multiple devices on the network before reaching the destination node and can be easily compromised if not secured. With such an importance of encryption, the network also requires a key management and distribution framework. As traditional network uses a centralized system for maintaining and distributing cryptographic keys in the network, it is a big challenge to implement the same in a distributed network with minimal dependence on a central entity. The key exchange must consider the nature of the network and accordingly incorporate improvements to be able to function in a distributed network. This thesis explores the above areas to propose a new network model for a secure dynamic wireless mesh network including a new routing scheme and a security framework comprising a hybrid encryption scheme, a hybrid authentication scheme and an improved key exchange and management scheme. This thesis demonstrates that our solutions not only strengthen and secure the dynamic wireless mesh networks but also significantly improve the performance and efficiency as compared to existing approaches.

Author's Publications

The author has published five refereed papers including one ERA ranked A[1] journal paper, two ERA ranked A conference papers and two ERA ranked B conference paper. The publications including one ERA ranked B conference paper that is under review are listed below in detail. The impact factor (IF)[2] of the journal paper is also stated.

Journal Article:

1. **Ashish Nanda**, Priyadarsi Nanda, Xiangjian He, Aruna Jamdagni, and Deepak Puthal, 2018, 'A hybrid encryption technique for Secure-GLOR: The adaptive secure routing protocol for dynamic wireless mesh networks', *Future Generation Computer Systems* (FGCS), https://doi.org/10.1016/j.future.2018.05.065, 2018 (ERA tier A, Impact Factor 4.639)

[1] ERA ranking is a ranking framework for publications in Australia. Refer to http://www.arc.gov.au/era/era_2010/archive/era_journal_list.htm for detailed ranking tiers. The 2010 version is used herein. For journal papers: A* (top 5%); A (next 15%). For conference papers (no A* rank): A (top 20%).

[2] IF: Impact Factor. Refer to http://wokinfo.com/essays/impact-factor/ for details.

Conference Papers:

2. **Ashish Nanda**, Priyadarsi Nanda, Xiangjian He and Deepak Puthal, 2018, 'A Novel Hybrid Authentication Model for Geo Location Oriented Routing in Dynamic Wireless Mesh Networks', 51st Hawaii International Conference on System Sciences(HICSS-51) Hawaii, 3rd-6th January 2018 – Published, (ERA tier A)

3. **Ashish Nanda**, Priyadarsi Nanda, Xiangjian He, Aruna Jamdagni, and Deepak Puthal, 2017, 'Secure-GLOR: An Adaptive Secure Routing Protocol for Dynamic Wireless Mesh Networks', 16th IEEE International Conference on Trust, Security and Privacy in Computing and Communications (IEEE TrustCom-17) Sydney, Australia, 1st-4th August 2017 – Published in IEEE Conference Proceedings by IEEE CS CPS, (ERA tier A)

4. **Ashish Nanda**, Priyadarsi Nanda and Xiangjian He, 2016, 'Geo-Location Oriented Routing Protocol for Smart Dynamic Mesh Network', 18th IEEE International Conference on High Performance Computing and Communications (HPCC-2016), Sydney, Australia, 12th-14th December 2016 – Published in IEEE Conference Proceedings by IEEE CS CPS, (ERA tier B)

5. **Ashish Nanda**, Priyadarsi Nanda, Xiangjian He and Aruna Jamdagni, 2016, 'A Secure Routing Scheme for Wireless Mesh Networks', 12th International Conference on Information Systems Security (ICISS-2016), Jaipur, India, 16th-20th December 2016 – Published in Springer Verlag series of Lecture Notes in Computer Science, (ERA tier B)

Table of Contents

List of Figures

List of Tables

List of Algorithms

Chapter 1

Introduction

In this chapter, we present an overview of the thesis. It provides the research background along with the research motivation. This chapter also presents a summary of the contribution made by the author. The chapter's structure begins with Section 1.1 which provides some background information on distributed networks and how they work, followed by the research goals including the challenges or routing and the security concerns. Section 1.2 provides the research motivation that led us to embark upon the development of a new secure network model. Section 1.3 provides a summary of the work conducted and lists the contributions made by the authors. Finally, Section 1.4 depicts the structure and organization of the thesis.

1.1 Aim and Research Goals

The need for a distributed network has been apparent more than a decade and various network models have been proposed and built to overcome these needs. As we achieve more advancements in technology, the number of electronic devices also increases along with it. Even though we have upgraded the traditional network models to cope with the consistent growth in the number of connected devices, these networks are still bound to a central entity. In addition to the constant upgrades, to expand the coverage area of such networks, it requires huge investments in the infrastructure that needs to be put in place [1, 2].

To address these challenges, we require a distributed wireless mesh network model which is easy to set up and requires minimal infrastructure to build and maintain. In addition, the network model should also address the various concerns a traditional network faces, such as the responsibilities of a central controller needs to be taken over by the devices that create the network themselves. In addition, the distributed network also requires adapting to the dynamic nature and mobility of the devices. The new network model must also address the security concerns related to such organization of the network.

1.1.1 Network Model

Mesh networks, in general, have been around for a while and play an important part in providing a communication medium in remote or hostile areas. The mesh network is used to relay data from electricity meters all the way to providing connectivity amongst the GPS-satellites [3-7]. Such networks have been designed to work independently without a central entity or control node and follow a bottom-up approach towards network formation. Unlike traditional networks where a central entity is in control of the network and must manage and maintain it, in a mesh

network the devices connect with each other to form the network and at the same time take the responsibility of managing and maintaining it. This is achieved by releasing the control from a central entity and handing it over to the devices that make the network. A wireless mesh network must also possess certain qualities making it a viable option to switch over from a traditional wired network that follow the current TCP/IP model for communication. These qualities can be distributed into three components:

1.1.1.1 Self-Sustainability

A wireless mesh network must always be self-sustainable, that is, be able to withstand the challenges of creating and maintaining a network by itself. It should not be dependent upon traditional infrastructure such as a broadcast centre or communication tower. The network should be able to utilize the resources available within the devices that enable it to achieve its full potential in order to sustain the network.

In a traditional network, the solution to sustainability is to keep upgrading and expanding the infrastructure to cope with new requirements [8]. This is however not a smart solution for a mesh network as it already has a pool of unused resources within the devices that make up the network. Given the structure of a mesh network, it is beneficial to have multiple devices contribute a small amount of resources balanced across the network than to use high amounts of resources at fixed points using few devices.

1.1.1.2 Scalability

One major concern with distributed systems is the extent of their scalability. Even though distributed networks are comparatively more scalable than traditional networks without setting up additional infrastructure, they do have a limited capacity. Given the distributed nature of the network, there is a predefined extent to which the network can add new devices as beyond that a saturation point is

reached resulting in the network breaking apart. This occurs due to the fact that in order to maintain a mesh network, each device must know the position of every other device on the network. This information is stored on each device in the network and is used to create routing paths when data is to be transmitted.

In order to make the network more scalable, a central node can be introduced which can take over the network mapping and route formation and can increase the number of active devices on the network at a given time. Although it's a viable solution, it makes the mesh network infrastructure dependant and loses its self-sustainability. Another possible solution is to introduce a database aided structure to minimize the amount of information each node needs to store and in the process, make the network more scalable.

1.1.1.3 Dynamic Support

The mesh network is not a new concept and has definitely been applied to various network scenarios before. However, the wireless mesh network is a relatively new addition and has made the transition from being a static network to a dynamic one. A wireless mesh network is ideal for mobile wireless devices but requires building a mechanism to incorporate the ever-changing nature of mobile devices.

As mobile devices are not stationary devices, they constantly move within the network shifting connections from one device to another. This creates the need to constantly keep updating the control information amongst the devices to be able to route data. The wireless mesh network must address this issue by incorporating the dynamic nature of the devices through major improvements in the network model structure itself.

1.1.2 Security Framework

Given the structure and organization of a wireless mesh network, implementing a security framework is both challenging and necessary. As the data travels through multiple devices to reach its destination, it can be compromised at various hops in the process. Although the traditional network model and the mesh network model share the same security feature set, the implementation is vastly different. Hence, from an implementation point of view, the security framework required for a mesh network can be classified as follows:

1.1.2.1 Authentication

In any given network the first and a very important line of defence is authentication. The main objective is to verify each user before they can connect to the network and to prevent unauthorized access to prohibited devices that mean harm. The concept is very straightforward for a traditional network which utilizes the central control entity for the verification process. This is however not the case with a distributed network like mesh given its expanded and ever-changing nature. The authentication scheme for a mesh network needs to adapt according to the resources available over the network as well as the resources present during a particular authentication scenario.

1.1.2.2 Encryption

Once a device has been authenticated, the second line of defence is provided by using end-to-end encryption. This important component of the security framework is responsible to maintain data integrity while it is being transmitted over the network. Achieving good encryption standard is highly dependent upon the type of encryption technique being used; that is, a stronger technique will provide better security. However, encryption has its own challenges when it comes to large dense network models as strong encryption techniques usually require a lot more

resources which may not be available in a mesh network. The encryption technique design for a mesh network must create less overhead by keeping the encryption process quick and less resource demanding while being able to provide a high level of security.

1.1.2.3 Key Management

A security framework is incomplete without a proper key distribution and management scheme. As all the cryptographic keys used for encryption and in some cases even authentication need to be generated and distributed in the network, the key management scheme must also achieve high levels of security. The most crucial component of the key management process in the initial key exchange where a device generates its key for the first time and sends it across the network to other devices that may wish to communicate securely. This, however, is also one of the weakest links as a rogue or malicious device can interfere with the process to interrupt it, steal the key or even replace the key. Hence, distributed networks such as the mesh require a much more decentralized system

1.2 Research Motivation

Wireless mesh networks have a lot of potential and can be used for various important aspects [9] due to their ease of setup and usage, such as:

- A wireless intercommunication system for industry [10].
    - The mesh network can be used to set up internal secure communication systems for industries without the need of additional infrastructure.
- Provide connectivity in remote areas [11].

    - As the mesh network does not require internet access to provide communication, it can be easily set up in remote areas to provide connectivity.
- Data collection over large areas [12, 13].
    - A large area of land can be monitored using various types of sensor modules equipped with near-field communication. By using a mesh network to connect such devices, they can transmit data over large distance automatically.
- Backup network in emergency scenarios like a natural disaster [14-16].
    - As a major concern in an emergency situation, such as natural disasters where the communication system is damaged or destroyed, is to maintain connectivity. The mesh network can be set up using portable devices, such as smartphones, and provide a backup communication network.
- Distributed Mobile Data Processing for complex computations [17].
    - The mesh network can be used to connect multiple high-performance devices together in order to divide complex computations into smaller chunks which can then be solved by individual devices on the network.

The performance of the wireless mesh network is, however, limited by the current network models that are applied. This is mainly because the current network models are modified versions of existing traditional network models. Due to the structural difference between traditional networks and mesh networks, the use of such modified network models limits the capabilities of a mesh network and bounds it to the specifications required by traditional networks.

The primary motivation of the research is to build a secure network model to improve and make the mesh network more functional. This work is comprised of creating a routing protocol designed exclusively for mesh networks along with a

security framework modelled around mesh networks to take advantage of the new possibilities. The routing protocols take into account the various factors missing in similar routing protocol and uses features specific to the type of devices that make up the mesh network. The routing protocol must convey availability, be self-sustained, scalable and should support the dynamic nature of the network. At the same time, the security framework must provide adequate integrity and confidentiality.

1.3 Research Overview

This section presents a summary of the research methodology incorporated in this thesis and our major contributions.

1.3.1 Methodology

In an effort to address the research challenges stated above, we have applied a comprehensive approach that begins with a study of existing distributed networks in order to explore their potential and to determine any possibility of improvement. In this phase, relevant network models are broken down into the routing schemes they use, the security they employ and the overall success in achieving the above-mentioned research goals. Each network model is then evaluated based on the possible features it can offer as compared to its current features. Using this data, we can pinpoint the limitations with current architecture and improve upon it. The steps involved can be summarized as below:

- **Collection:** This step involves collecting relevant information from various sources related to the research in question

- **Evaluation:** Once relevant information is collected, it is evaluated based on certain parameters to check its usability.
- **Collation:** Once the evaluation phase is complete, the collected information is organized based on different aspects of the network model
- **Analysis:** Each piece on information is analysed to provide important information regarding its role and working.
- **Dissemination:** All the information is then distributed amongst the various key factors they impact.
- **Resolution:** Using the information collected, each key factor is redesigned or modified based on its assessment.
- **Companionable:** The new key factors are analysed to verify their compatibility with each other as well as the research goals to build a possible solution
- **Experimentation:** The solution is then tested using a simulation of a test environment and/or a theoretical analysis based on its nature.

1.3.2 Contributions

Our contributions, presented in this thesis, are mainly divided into 4 chapters; each one is discussed in short as below:

1. One of the major concerns with a distributed network, as discussed above, is the lack of distributed network specific routing protocols. The routing protocols that do exist are the ones that have been ported over from traditional networks that are more static oriented and restrain networks such as the mesh from achieving its true potential. The proposed routing protocol, Geo-Location Oriented Routing (GLOR), has been designed for dynamic wireless mesh networks and features scalability, sustainability, and dynamic support. The routing protocol is built around the concept of both

control and resource decentralization and does not limit the expansion of mesh networks. The proposed routing protocol also includes changes to the network topology itself enabling a much more flexible mesh network model. This contribution is based on our publications [18, 19].

2. As with any network, security is a major component, hence we propose the Secure-GLOR network model comprising a security framework applied to the GLOR protocol. The security framework has been divided into three components with the first one being authentication. As mentioned before, implementing authentication in a mesh network is difficult due to the missing central entity. Hence, keeping the requirements of a distributed network in mind, we propose the hybrid authentication scheme which works based on the different scenarios of the authentication process. The proposed scheme is aided by a central database but does not completely rely on it. This work is based on our publication [20].
3. As the second component of the security framework, the next contribution is the hybrid encryption scheme for distributed networks. The proposed encryption scheme addresses adequate resource utilization along with using a combination of symmetric and asymmetric encryption techniques. This helps in minimizing the resource requirements along with lowering the network overhead caused due to the comparatively large size of the encrypted files. The scheme is based on our publications [21, 22].
4. The third component of the security framework is key management and distribution for distributed networks. In the proposed scheme, the authors have undertaken a major concern with key distribution schemes in distributed networks. The key distribution is an essential part of the framework as the security of the network can be easily compromised if the key exchange is not secure itself. The proposed scheme takes into account the complexity and probability of a crypto key being generated and uses

concepts such as randomization and anonymity to prevent any malicious devices from interrupting or compromising the crypto keys.

1.4 Thesis Organization

This thesis has been organized as below:

- Chapter 2 presents the background and related works relevant to the research area. The chapter discusses the mesh network, existing distributed network models, routing techniques and protocols. The chapter also discusses the various security frameworks for distributed systems and how they apply the security. The chapter elaborates upon the working and the limitations of the above-mentioned network models and security frameworks
- Chapter 3 presents our proposed Geo-Location Oriented Routing protocol (GLOR) and discusses in detail its various components and workings. The chapter sheds light on how the proposed GLOR protocol addresses the various concerns and limitations related to current routing schemes. It also presents the various new features of the protocol along with how they are incorporated to provide extended functionality. The Secure-GLOR security framework is also briefly discussed.
- Chapter 4 focuses on the authentication component of the security framework and presents the hybrid authentication scheme. The chapter discusses the various components of the scheme and how they are integrated into the mesh network. It also discusses the various scenarios the proposed authentication scheme can handle.
- Chapter 5 is dedicated towards assessing current encryption/decryption schemes and testing its performance over the mesh network to find a

suitable solution towards reducing the resource requirement and the network overhead. The chapter discusses the benefits and flaws of symmetric and asymmetric encryption and how it can be used in a combination to provide a viable solution.

- Chapter 6 provides the final component for the security framework and focuses on key management and distribution. The proposed MPARK distribution scheme is discussed in detail with its various components and how they reduce the chances of the key exchange being compromised by both internal and external factors.
- Chapter 7 concludes the thesis and provides a window into the future works possible in improving and expanding the presented Secure-GLOR network mode.

Chapter 2

Background Studies and Related Works

The mesh network has been under development for a very long time; however, most of the work includes adding features to existing network models and applying that to the mesh. In relations to the proposed security schemes for the mesh network, most imply a centralization of the various functions. This chapter discusses the current network models and security implications for the mesh network. Section 2.1 provides a general overview of the mesh network and its security needs. Section 2.2 discusses data transmission techniques followed by a review of existing routing protocols in Section 2.3. Relevant network models are discussed in Section 2.4. Section 2.5 sheds light over various security threats that apply to the distributed networks. This is followed by a discussion of key security aspects that impact the security framework in Section 2.6. Finally, Section 2.7 summarizes the chapter.

2.1 Research Overview

Wireless Mesh Network [23-27] is an emerging technology with great potential to become a self-sustained network. Unlike the traditional networks that dominate the communication system and rely on large and expensive setup of wired/wireless access points to provide a connection between users, Wireless Mesh Network is formed by the user devices which connect to each other to form a network as portrayed in Figure 2.1.

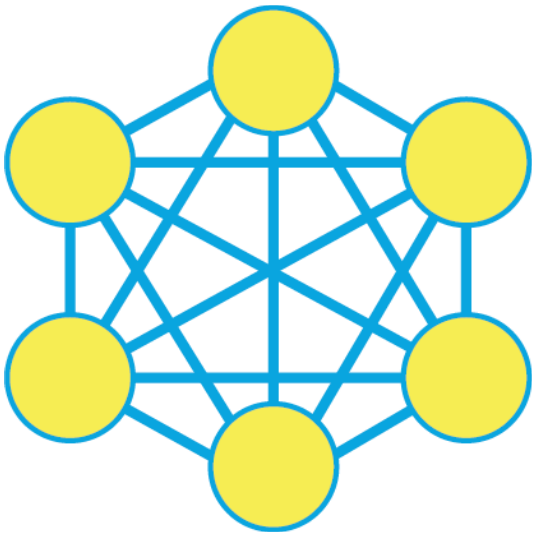

Figure 2.1 Mesh Network Topology

The wireless mesh Networks are known for their reliability as they are formed by several connected devices (nodes) through which the messages are relayed using either a flooding technique or a routing technique. This is achieved by hopping the message from one node to another until it reaches the destination as shown in Figure 2.2. They also have the ability of self-healing, allowing a routing-based network to operate when a node breaks down or when a connection becomes unreliable by automatically creating a new one as shown in Figure 2.3.

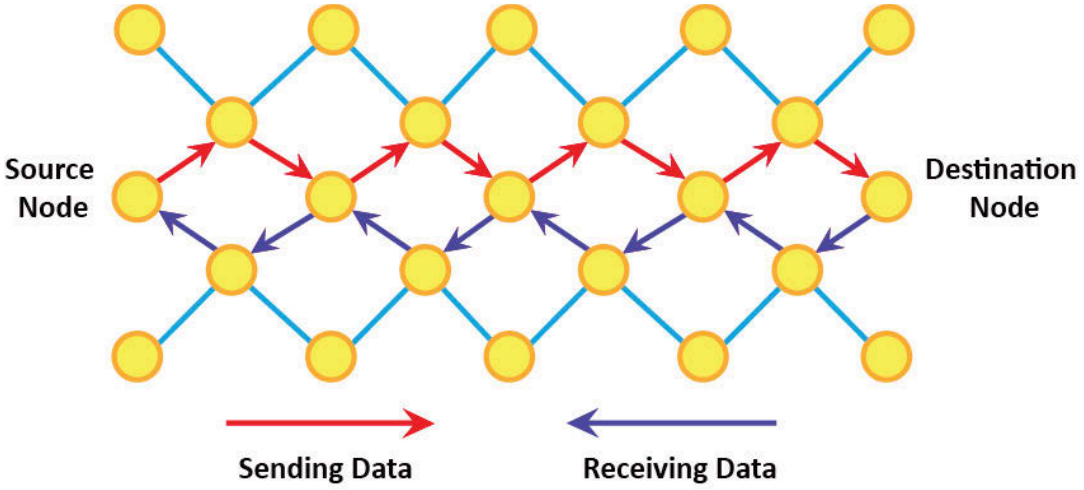

Figure 2.2 Data Transmission in a Mesh Network

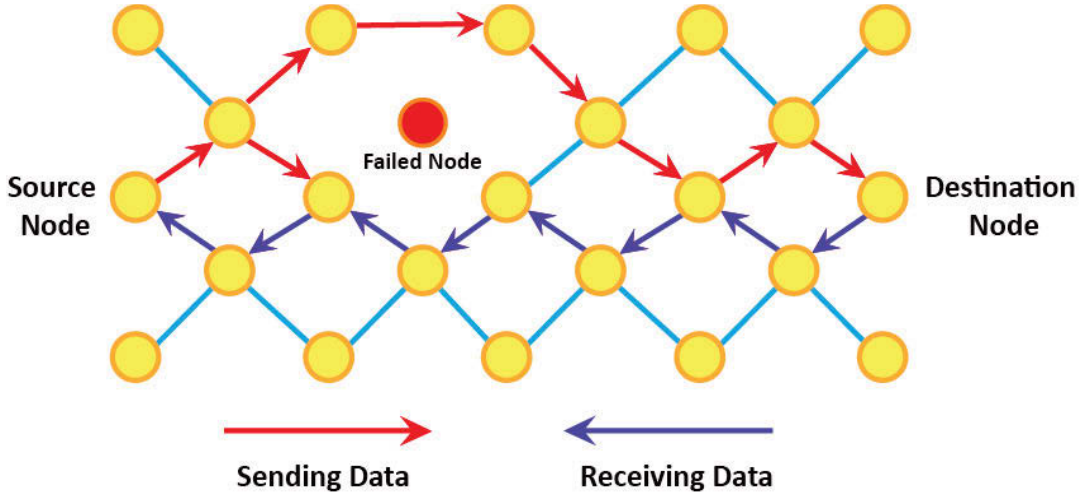

Figure 2.3 Self-Healing in a Mesh Network

Wireless Mesh Networks specifically have been around for a while and have been used to build distributed networks, connect satellites to enable satellite calling and even for data collection from electricity meters spread across wide areas [3, 5-7]. The Mesh Network has been under examination and experimentation to achieve a network model that is self-sustained, secured, scalable and dynamic. In the past few years, researchers have realized that the mesh networks hold the potential of becoming the network of the future, however, only a few attempts have been made to achieve it.

The current versions/implementations of the wireless mesh network face several challenges. One of the major problems arises due to the fact that the signal is rerouted and hopped from one node to another and there is a limit to the number of nodes it can have. If the number of nodes increases, a central controller/access point is required to control the network which on failure compromises the network's connectivity. Another problem arises with the dynamic nature of the mobility of the devices.

Due to the use of legacy/traditional approaches on mesh networks, as explained further in Section 2.4, there exist various limitations towards its implementation. As most of the protocols that exist for mesh networks are modifications of the previous static protocols, there are various limitations to the dynamicity on the network they support. These protocols also use the legacy system for device identification which is not suited to mesh networks as the devices are mobile and continuously switch between connections.

Another issue arises with the overhead traffic, as most legacy protocols first map the network through TC/Hello packets in order to create routes. When the protocol is modified for mesh networks, this causes a lot of issues as the mesh network is dynamic and the network map is constantly changing. This leads to a lot of overhead trying to map the network which again results in more processing and power consumption. The network overhead increases with the number of devices and after a limit, it causes the network to self-saturate.

As with any other communication network, security is a very important concern in wireless mesh networks as well. With all the data travelling openly through multiple devices, it is a challenge to maintain the privacy and integrity of the data. As discussed at various instances before, the wireless mesh network's unique features require a security framework tailored to it with security schemes that complement the open nature of the network. In the coming sections, we discuss in

detail the various routing models that exist, their various components, how they work and the limitation they put forward. We also discuss existing security frameworks designed for distributed networks and how they stand against various security threats. In addition, the security frameworks are also broken down into their components to discuss their workings and limitations.

2.2 Data Transmission Techniques

In any given mesh network, the routing protocol is responsible for the identification of devices on the network, their addition to the mesh network and the transmission of data from one device to another within the network [28]. The data can be transmitted in a mesh network using one of the two ways discussed below. In some cases, a combination of the two techniques may also be used.

2.2.1 Flooding - Broadcast Technique

The flooding or broadcast technique, as the name implies, works on the concept of distributing data to every device without any direction. This is achieved by having every device in the network re-transmit each piece of data to each of its connected neighbour devices, who in turn will do the same. This re-transmission process is repeated multiple times until every device on the network has received the data packet, implying that if the destination was a part of the network during the broadcast, it should have received the data as well. Figure 2.4 depicts a sample broadcast from a source device to the destination device.

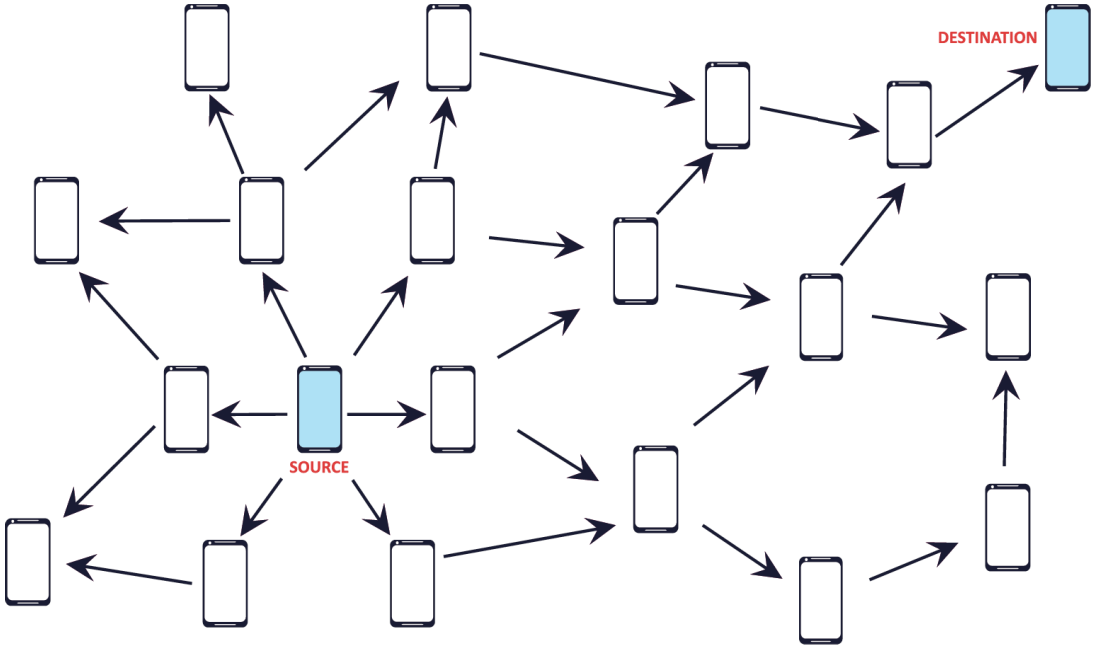

Figure 2.4 Flooding / Broadcast Technique Sample

Limitations

As seen in the figure, each black line denotes the data transmission and can be seen originating at the source node and diverging outwards throughout the network. The technique works very well for small sized networks, however as the size of the network increases so does the network overhead [29]. This also increases the network load on individual devices by having up to 90% of the data transmissions amount to excessive resource consumption.

In addition to the above-stated limitation, the broadcast technique cannot be used for individual device identification, hence the communication in the network is open and each data packet sent out is received and read by every other device on the network. Such network structures pose a great threat and can result is a compromised network.

2.2.2 Unicast - Multicast Technique

Unlike the broadcast technique explained above, the Unicast or Multicast technique uses a pre-defined path through which a packet is sent. The unicast method uses a single path whereas a multicast method can use 2 or more paths for data transmission (as depicted in Figure 2.5), providing redundancy in case one of the paths dies out.

The routes used for data transmission are calculated using a neighbour table containing the location of each device on the network. These devices create their neighbour tables by distributing 'Hello' packets across the network and compiling the responses they get from other devices on the network. Using this data, a device can calculate the path and number of hops it would take for the data to reach its destination.

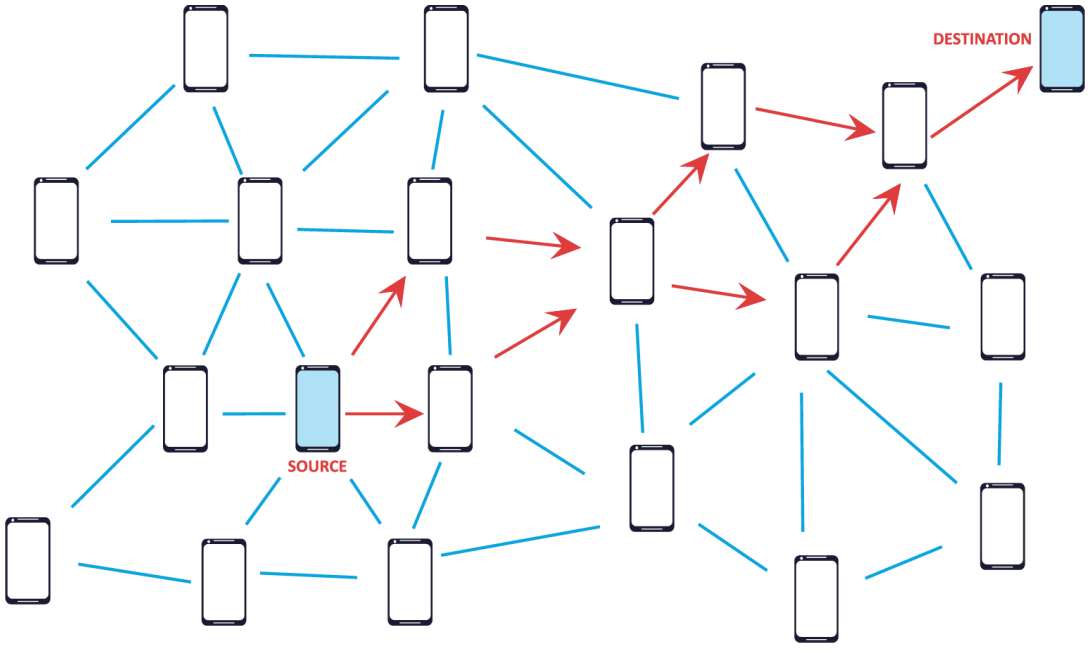

Figure 2.5 Multicast Routing

Limitations

The network mapping method used for this technique is however responsible to saturate the network itself. As the number of devices on the network increases, so does the number of 'Hello' packets travelling across the network leading to an uncontrollable number of responses resulting in network saturation. Given this limitation, it becomes difficult to expand the network. In addition, using mobile devices increases the network instability as, given the dynamic nature of the mobile devices that can change their location, this leads to ever-changing transmission paths altering data flow.

One possible solution to support a larger number of devices requires a central entity that can store the network information, such as device location, and can hence calculate data transmission routes for other devices [30]. This approach, however, compromises the very basic principle of a mesh network: its ability to be self-sustained, thus making the network prone to failure in case the central entity was to fail resulting in ceased data transmissions and device isolation.

2.3 Routing Protocols

Since the introduction of the distributed network topologies, a few routing protocols have been developed and various others have been ported over from the traditional networks and accordingly modified to be compatible with the distributed network topology. In this section, we will discuss the most relevant and commonly used routing protocols for distributed networks [31-35]. In addition to the working and its features, the section will also discuss the limitations faced by the routing protocol.

2.3.1 Optimized Link State Routing (OLSR)

OLSR is an optimization of the classical link state algorithm design, modified in accordance with the requirements of a mobile wireless LAN [32, 36-45]. The first optimization achieved is due to the key concept used in the protocol based on Multi-Point Relays (MPRs). MPRs are selected devices (nodes) which forward broadcast messages during the flooding process [37, 46, 47]. This technique substantially reduces the message overhead as compared to a classical flooding mechanism, where every node retransmits each message when it receives the first copy of the message.

In OLSR, link state information is generated only by nodes elected as MPRs, thus a second optimization is achieved by minimizing the number of control messages flooded in the network. As a third optimization, an MPR node may choose to report only links between itself and its MPR selectors. Hence, as contrary to the classic link state algorithm, partial link state information is distributed in the network. This information is then used for route calculation. Each node in the network uses a routeing table that stores the information regarding the path and hops required to reach any other node on the network.

OLSR provides optimal routes (in terms of the number of hops required to transmit the data). The protocol is particularly suitable for large and dense networks as the technique of MPRs works well in this context. The Protocol has been evolving since its introduction to incorporate new challenges that are faced in a mesh network. The OLSR version 2 [48] was released to incorporate such aspects and to provide a better experience for a wireless dynamic mesh network.

Limitations

The OLSR protocol was initially ported over from the traditional link state protocol and was designed to handle static networks. Although over the years

various developments and alterations have made the OLSR protocol applicable to wireless dynamic mesh networks, it is unable to handle an ever-changing network optimally. In addition, as the network work on a hierarchy system with certain nodes in the network having more responsibility than others, it creates weak links in the network making it susceptible to failure should an MPR node fail or is compromised. The protocol is also known to cause self-saturation on the network as even though it is optimized to reduce the 'Hello' packets travelling through the network, after a certain limit of devices is reached it becomes unstable and unable to maintain and map the entire network efficiently.

2.3.2 Ad hoc On-Demand Distance Vector (AODV) Routing

The Ad hoc On-demand Distance Vector (AODV) routing protocol is intended for use by mobile nodes in an ad hoc network [32, 38, 39, 49-56]. It offers quick adaptation to dynamic link conditions, low processing and memory overhead, low network utilization, and determines unicast routes to destinations within the ad hoc network. It uses destination sequence numbers to ensure loop freedom at all times (even in the face of anomalous delivery of routing control messages), avoiding problems (such as 'counting to infinity') associated with classical distance vector protocols.

Similar to the OLSR protocol, AODV routing protocol uses periodic 'Hello' packets being broadcasted by every node in the network but this is only limited to its neighbour nodes. This helps in reducing the overhead caused due to the flood of 'Hello' packets. In addition to the reduced overhead, given the 'Hello' packets are periodically sent, if a node fails, its neighbours will know of it early and can redirect the traffic through a different link. The routes are created on-demand initiated by a Route Request (RREQ). When a given node needs to transmit data, it will broadcast an RREQ to its neighbours. The neighbours will do the same and keep on saving the

hops it takes to finally reach the destination node thereby creating a route for the data to be transmitted through.

Limitations

The protocol performs very well with a smaller network size; however, with an increase in the number of devices the protocol starts to slow down. Given the structure for finding routes depends on RREQs, with the increased size of a network the RREQ takes a longer time to find viable routes for the data to travel. In addition, the routing protocol is unable to handle large dynamic networks with a majority of mobile devices as the network topology keeps changing. The protocol finds it difficult to keep up with the constantly changing connections between moving devices and hence route creation becomes, even more, slower leading to discovered routes expiring before the data can even be sent across the network.

2.3.3 Zone Routing Protocol (ZRP)

The Zone Routing Protocol (ZRP) works alongside a packet delivery service referred to as the Bordercast Routing Protocol (BRP) [57-59]. ZPR is a good example of a hybrid reactive/proactive routing protocol. The essential working of this routing technique is based upon routing zones, wherein each node must maintain a link with all other nodes within a pre-defined number of hops. As any given node is only required to remember the routes for a small portion of the network, it does not overload the network by having each node map every other node. In addition to the reduced overhead, given the small area a node must remember, it is easier for a node to be updated about its neighbours and keep track of changing routes and moving devices.

The data being transmitted from the source node within its zone is handled proactively; however, if the destination node lies outside the source node's zone, it

is handled by a reactive protocol. The reactive protocol used is BRP which maintains that any given packet is transmitted to the peripheral nodes which will then proactively search for a route within their zone and the process repeats until a viable route is found to the destination node.

Limitations

The ZRP protocol has many desirable features and is truly one of the first hybrid protocols, however, it still relies on searching and establishing routes before data can be sent. This interferes with the dynamic nature of the network especially when a given destination does not exist in the same zone as the source. Hence the protocol is more suited to the static nature networks. The ZRP also works in conjunction with other routing protocols and is dependent on other protocols to work properly in order for it to increase efficiency. The fact that it is dependent on other protocols, it is also limited by their limitations and cannot be scaled beyond their limitations.

2.3.4 Destination-Sequenced Distance-Vector (DSDV) Routing

Destination-Sequenced Distance-Vector (DSDV) routing protocol, known to be an older version of AODV routing protocol, can implement both a link-state and distance-vector approach [38, 55, 60, 61]. The DSDV routing protocol uses tables to store the routing information, similar to the OLSR protocol. Each node on the network maintains a record of every other node on the network using a routing table. The table provides information regarding the neighbour node to be selected as the next hop for each possible node in the network. This table is created using two types of packets, full dump and incremental. A full dump packet carries all the hop information for the entire network as is broadcasted frequently, whereas an incremental packet is only sent when the network changes and contains only the changes that have taken after the full dump packet was sent. This is used to lower the overhead on the network.

Limitations

As with any routing protocol using routing tables, the DSDV routing protocol cannot handle dynamic networks comprised of mobile nodes. This is because of the constantly changing network structure causing the information stored in the routing table to become invalid and in need of constant updating. In addition, the network will be overrun by incremental packets as a dynamic network is constantly changing.

2.3.5 Dynamic Source Routing (DSR)

Dynamic Source Routing (DSR) uses the concept of source routing using route cache which stores the route information required to transmit a packet [54, 55, 59, 62]. The technique does not use any periodic packets for updating route cache, instead, each packet being sent over the network updates the information. The routing technique has two important components, route discovery and route maintenance, both of which are essential and often work together.

When a source node is sending a packet, it stores the information about each hop of the network into the packet header. This information is used by the next hop device to determine its next hop or if it has reached the destination. In case the next hop is unable to locate the next hop, it will initiate a route request resulting in updating of the route cache which will then be used to route the packet. Using such techniques, the DSR protocol reduces the bandwidth overhead helping in conserving battery and avoiding large routing updates. In addition to the above benefits, as the protocol does not rely on pre-set routes, a discovered route is always the shortest route possible given the route request originated from the source node and the destination node was the one to respond.

Limitation

In order to reduce the load on each node created by regular network update packets, DSR allows all nodes on the network to snoop on all the network traffic that passes through them, including the data packets and they store the complete route information. This potential feature can turn into a major flaw if some packets with wrong routes were distributed by a malicious node and could end up destroying the network. In addition, every node on the network is able to snoop upon all the traffic and is a major security threat. This also implies that if a security framework was to be employed to address the security concerns, the routing protocol would fail as snooping would not be possible.

2.3.6 Temporally Ordered Routing Algorithm (TORA)

The Temporally Ordered Routing Algorithm is a distributed routing protocol known to be highly adaptive, efficient and scalable [38, 54, 55, 59, 63]. TORA is a hybrid reactive multicast protocol, also referred to as the link reversal protocol, that only finds a route after a source node has initiated a request. The protocol also uses the concept of link reversal and expects the nodes to remember their neighbours.

The working of their routing protocols is governed by three components: create the routes from source to destination, maintain the routes and erase invalid routes. When a source node initiates a route request, the network will look for multiple routes to the destination node and keep a record of them. This is to avoid having to look for new routes every time a request is initiated. The network keeps checking the routes to maintain their connectivity and does not allow a node to initiate a route request until all the available routes have been erased. In an event where the network partitions, the protocol erases all existing routes.

When the source node does not have any viable routes present, it initiates a route query. The routing information is gathered by broadcasting a query packet to the neighbouring nodes, who in turn do the same until a route to the destination node is found. However, unlike other routing protocols, TORA accepts all possible routes instead of the most viable one. Once the routes are discovered, the node can use one or more paths to send the data and has to maintain a continuous downstream to maintain the connection.

Limitations

TORA routing protocol does face some limitations, an important one being the time synchronization. As the network depends upon the synchronized timings of each of the nodes in the network, a small difference in the time sync can result in big changes. In addition to the time sync issue, the network protocol does not address important components such as link status, neighbour discovery and address resolution; it requires an underlying protocol to do these tasks. This makes the TORA protocol dependent upon the underlying protocol and bound by its limitations as well in addition to its own.

2.4 Network Models

There exist various network models and schemes for distributed networks that encompass the entire workings of the network, however, only a few of the existing networks exhibit a de-centralized network accurately. In this section, we study the most relevant network models and their suggested approach for a truly distributed network. Each network model is then analysed for its unique features, its working, implementations along with the limitations it faces.

2.4.1 The SPAN Project

This particular approach was the first practical implementation of Mesh Network to work off-grid introduced in the year 2012 [64]. The approach, known as the Smart Phone Ad-hoc Networks (SPAN), uses the existing hardware (primarily Bluetooth and Wi-Fi) in commercially available smartphones to create peer-to-peer networks without relying on cellular carrier networks, wireless access points, or traditional network infrastructure. SPAN differs from the traditional hub and spoke networks, such as Wi-Fi Direct because it supports multi-hop relays and there is no notion of a group leader so peers can join and leave at will without destroying the network.

Implementation

The implementation of this approach involves injecting a MANET (Mobile Ad-hoc Network) [65-67] based framework between OSI layer 2 & 3 (Data link & Network). This allows the arbitrary routing and implementation of custom protocols. The test network created is able to implement chat/text conversation in a 5-hop network with a minimal delay. The initial size of the network was set to 30 devices and the average distance for good connection was found to be 30 meters. The routing technique used to implement the network was OLSR (Optimized Link State Routing) which had been modified to support such scenarios.

Security

According to the network model laid out by the authors, the SPAN network implements security using asymmetric encryption technique where a set of public and private keys are generated from which the public key is shared during first interaction between two devices in the network.

Limitations

The SPAN network model, being an early introduction to the distributed network approaches, does have various limitations. A major limitation, highlighted by the authors themselves, explains how the protocol itself can saturate the network with an excess of hello packets distributed to build neighbour tables during normal operation. Such concerns can also be used by malicious attackers to deliberately saturate the network. In addition to previously stated major flaws, the approach is more focused towards using a device with internet access to provide a gateway to the internet for other devices in the network that are unable to do so.

The implementation also features communication channel allocation based on individual devices which limits the maximum number of devices that could exist in the network at the same time and communicate with each other. During the tests it was also found that only the smartphones equipped with a certain type of communication chipset were able to implement the framework, limiting the support for a wide range of devices equipped with a different chipset.

2.4.2 The Several Project

The Several Project was created in 2010 in response to the Haiti Earthquake and has evolved and been upgraded ever since [68-71]. The primary objective of this project is to provide infrastructure for a backup and recovery network allowing direct connections between cellular phones through their Wi-Fi interfaces, without the need for a mobile phone operator. The approach features live calling as long as the mesh network can find a stable route to the destination node. Sending text or other media is achieved using the concept of Rhizome, a delay-tolerant data transferring concept for long distances with unpredictable connectivity. The Several project is essentially a set of protocols and technologies brought together in order to provide

infrastructure-less communication featuring important services oriented towards an after-disaster scenario.

Implementation

Over various iterations and improvements, the Several project has produced a viable option for a stand-alone network along with a collaborative mapping application intended to support disaster relief and recovery efforts. In addition to the application, the Several project also introduces a hardware referred to as the Several Mesh Extender [11] (Shown in Figure 2.6) used to establish a short-range Several mesh over Wi-Fi and join it with other more distant mesh networks by linking with other Several Mesh Extenders over packet radio operating in the ISM 915 MHz band.

A demo of this approach has also been conducted using an environment which was designed to implement an after-earthquake scenario. Various mobile devices were randomly placed around the complex and a demo rescue mission was showcased. The network was successfully formed by the devices spread around the complex, the trapped victim could easily contact the rescue personnel and the victim's location was also triangulated using the network.

Security

The Several approach incorporates end-to-end security using asymmetric encryption. The approach specifies a 256-bit encryption key through which individual devices are identified; these keys are referred to as SID (Several ID). In addition, the scheme also addresses confidentiality, integrity and authenticity without the use of a trusted third party. The overall security provided is also tailored to support the Rhizome technology to facilitate availability.

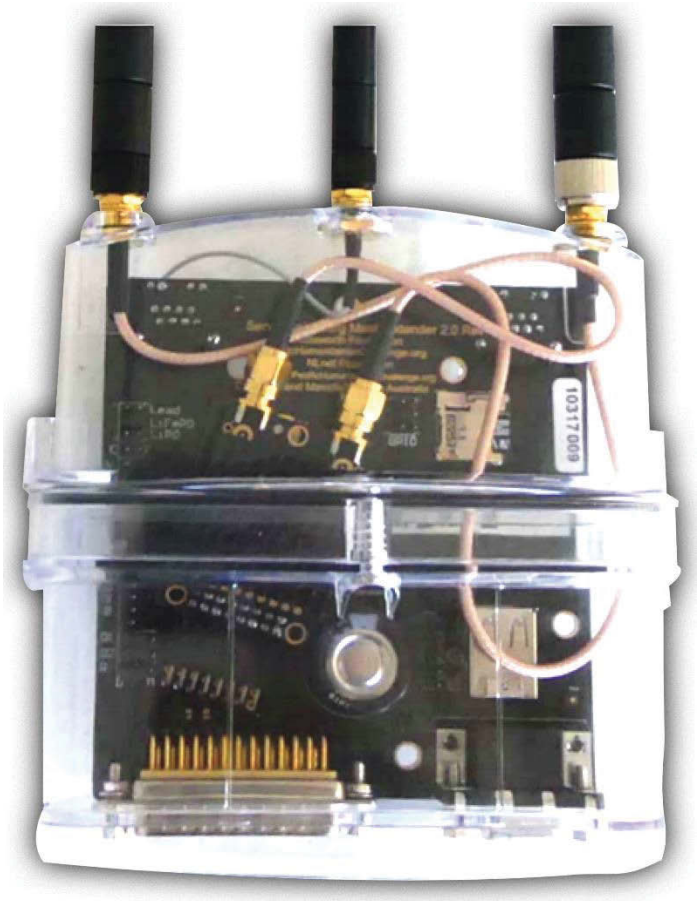

Figure 2.6 Generation III Several Mesh Extender [11]

Limitations

The Several network concept provides various services without the need of a cellular network; however, it does require the setup of another hardware dependent infrastructure, the Several Mesh Extender, in order to provide the mentioned services. Use of such devices may increase the range of the network but limits its dynamic-ness and makes the network partially infrastructure dependent. In addition to the partial infrastructure dependence, the Several approach also uses the Rhizome system, a type of Delay Tolerant Network (DTN). A DTN approach addresses the difficulty in transmitting data within a network that does not feature a route certainty [72]. Hence in a DTN when a device receives some data to be forwarded, it

is not obliged to do so immediately. The device can store the data until it is able to find a possible next-hop for the destination device which slows down the exchange of information and increases the data collection threshold for each device, which in a normal network would have forwarded the data directly.

2.4.3 Open Garden: FireChat

Open Garden is an organization that offers software solutions, one of which is a proprietary internet community-based connection sharing software application that is used to share internet access with other devices over Wi-Fi or Bluetooth [73]. This is achieved by creating devices with internet connectivity as gateways and setting up localized networks using Wi-Fi or Bluetooth to share the access to other devices lacking a direct access by using the Open Garden's application on their device.

The created network keeps analysing the network to avoid potential drops by switching between possible gateways to provide uninterrupted internet access to the network by introducing a way to access the Internet over multiple channels at one time, improving speed and reliability.

Implementation

Open Garden implements its network model through the application FireChat and has approximately 5 million downloads so far. The application became very popular during various musical events or even during riots to provide connectivity. It works using the broadcast routing method under which a single message is relayed to multiple devices connected to each other in the hope of making it to the destination device eventually.

The application features various types of modes which enable a user to choose the proximity, range and number of devices that you wish to communicate with. The application's design for off-grid communications only works on three modes:

- Everyone mode - It features a chat room where all the users in a given proximity can chat with each other.
- Nearby mode - It allows users to find other users close by.
- Firechat mode - It is a common chat room with all connected users talking about the same topic.

Security

The Open Garden's application FireChat does not mention the use of any security framework, however, when analysed it shows a basic implementation of user registration and authentication.

Limitations

Despite the number of downloads for the application, it has various limitations. To start with, the network's primary goal is towards providing internet access and is hence dependent upon the access gateways for the internet connectivity. Another major concern rises with the use of the application as a user intending to use the application must first download and install it followed by a user registration in order to be able to use it. As both these tasks require an internet connection, the network will not be able to work in stand-alone mode.

In addition to specific requirements required for the application to work, the network uses a broadcast method for data transmission. As each piece of information shared is sent to every device on the network, it increases the overhead and reduces the efficiency of the network model in accordance with resource utilization. Another major flaw is the missing security features as any data sent over the network is sent as plain text, any device on the network can view it as making

the network very un-secure. Only recently has the option to use end-to-end encryption for communication between users has been added. The application is also unable to identify/differentiate between different nodes and treats all as same.

2.4.4 The BRIAR Project

Briar was started by Michael Rogers as open source software for mesh networking technology. It is intended to provide secure and resilient peer-to-peer communications with no centralized servers and minimal reliance on external infrastructure. The connections are made through Bluetooth, Wi-Fi, or over the internet via TOR and all private communication is end-to-end encrypted. Any relevant content is stored in encrypted form on participating devices.

Unlike traditional messaging tools such as email, Twitter or Telegram, the Briar application doesn't rely on a central server, the messages are synchronized directly between the users' devices. Briar also provides private messaging and public forums that are protected against surveillance and censorship threats [74].

Implementation and Security

Briar is referred to as a delay-tolerant social overlay by the creators that offers a very high level of security. The explanation for each component is as below:

Delay-tolerant: Data doesn't need to travel from its origin to its destination instantly. When a device receives data, it may store it and pass it on at another time and place. This allows Briar to operate under more challenging conditions than mesh networks, which require a path from the origin to the destination to be found in real time.

Social: Briar uses the existing offline trust relationships between users to bootstrap secure connections. Devices don't communicate directly unless their

owners are contacts, and they don't broadcast the fact that they're running Briar, so eavesdroppers can't easily discover or enumerate Briar users.

Overlay: Briar can operate over a wide range of transports, including Bluetooth, Wi-Fi, Tor, dial-up modems and even USB sticks. Briar knits together these diverse transports to create an overlay network that doesn't depend on any single transport. The protocol stack provides the same security guarantees regardless of the transport, and new transports can be added quickly in response to developments in censorship.

Limitations

Similar to the Several Project, this network model follows Delay Tolerant Networking system which concludes that the data does not need to travel from its origin to its destination instantly; once a device has received data, it may store it and pass it on at another time and place when a connection is available. In order to implement high levels of security, the devices don't communicate directly unless their owners are common contacts, it means that a device ‘A’ can communicate with a device ‘C’ through another device ‘B’ only if the device ‘A’ and device ‘C’ exist as contacts on device ‘B’. This makes it difficult for the network to expand or improve functionality. Finally, Briar operates at the application layer rather than the network layer, so it is not possible to run existing TCP/IP applications on top of it as is the case with a mesh network.

2.5 Security Concerns in Distributed Networks

As discussed in the previous sections, security is an essential part of any communication network. Distributed networks, however, require security to ensure proper working of the network [42, 75-80]. This is due to the distributed nature of

the network as compared to the centralized one in traditional networks. The centralized networks can sustain security attacks as long as the central control entity is protected as compared to a distributed network where the control is distributed and the network will be adversely affected if a significant number of devices are affected by the attack.

The security concerns relevant to distributed networks, specifically the mesh network, consist of both existing threats and new threats given the topology. These threats can be classified under Availability, Authenticity, Integrity, Confidentiality, Non-repudiation and Anonymity [78, 81-91]. Each of the classifications has been discussed in detail below and includes the importance of the classification to the security model, the various threats that are a part of it and the effect of certain attacks on the overall workings of the network.

2.5.1 Availability

The availability of the network refers to the survivability of the network in case an attack happens. Even though availability relates more towards network performance, it is considered an integral part of the overall security of the network and requires certain fail-safes to ensure network sustenance. A distributed network is susceptible to certain attacks that can affect the communication medium itself and cause major damages to the connectivity. Some relevant attacks are as follows:

2.5.1.1 Signal Jamming

This type of attack usually takes place on a physical level and disrupts the physical and media access control layers. The magnitude of the attack can range from interfering with certain hardware components causing certain devices to disconnect all the way to jamming entire frequency bands leading to entire network failure. Even though intentionally jamming any signals is against the law, signals can still be affected unintentionally by certain equipment such as high-powered machinery.

2.5.1.2 Distributed / Denial of Service (D/DoS)

A distributed denial of service (DDoS) or denial of service (DoS) is a major threat for mesh networks as it can affect any layer of the network. The attack can be initiated by one or more devices sending out a large number of packets in order to interfere with the normal workings of the network. These packets can be of multiple types starting with 'Hello' packets used for network mapping, wrong information about possible routes, all the way to random data packets being sent across to overload the network. If no action is taken against such attacks, they can lead to a lot of overhead resulting in network saturation.

2.5.1.3 Sleep Deprivation Attack

The main objective of this attack is to target an important physical resource of the mesh network, that is the battery and hence is also referred to as the battery exhaustion attack. As mesh networks mainly comprise of mobile devices with a limited amount of power at their disposal, this type of attack can cause interference to the performance causing a rapid battery drain.

Once a device battery drains, it is unable to stay a part of the network and falls out. When enough devices fall out, the network has to struggle to maintain connectivity and finally breaks down. The attack can be initiated by simply overloading the tasks for a certain device or a group of devices leading to more power consumption.

2.5.1.4 Sink Hole and Black Hole attack

These types of attacks are initiated by rogue or compromised network devices by redirecting the network traffic towards them. The objective of a sinkhole attack is to tamper with or redirect the network flow whereas a black hole attack simply acquires and then discards as much network traffic as it can. Both the attacks can cause significant delays in the network by damaging the transmission routes and failing to deliver the data resulting is repetitive transmissions.

2.5.2 Authenticity

The authenticity in a communication is defined as the ability of a node to be able to verify the identity of another node it wishes to communicate with. This concern is handled by a traditional network by setting up authorization nodes and central control units. However, as a mesh network does not usually come with such components, it requires a different approach with an adaptable solution. Some of the most relevant threats that pertain to authenticity are as follows:

2.5.2.1 Impersonation

As the name suggests, this type of attack involves an attacker or malicious user impersonating a user node or even a control node. The aim of the attack is to divert any traffic from the actual destination node towards the malicious one by forging the identity of a user node. The attack can also be used to interfere with or sabotage control packets if a malicious node is able to impersonate a control node or a parent node.

2.5.2.2 Rogue / Compromised Device

If one of the existing devices on the network is compromised by an attacker through physical or remote means, it can be used to monitor the network traffic and in some situations, can also be used to distribute malicious content. In addition to the device getting compromised, a user may also go rogue and can cause the network significant damage.

2.5.3 Integrity

The integrity of the network is another important aspect that ensures that any packet transmitted through the network reaches the destination without being tampered with. An integrity check is a guarantee provided by the network to the

source node and the destination node that safeguards the data during transmission from any alterations, be it intentional or unintentional. The integrity of a message can be compromised in two ways:

2.5.3.1 Malicious Altering

This type of altering is achieved by using any type of attack on the network that provides access to the network traffic and is referred to as intentional altering. Once a malicious entity gains access to the network traffic, it can capture the data packets being transmitted and alter them. The alteration can be done in various ways, such as replacing the contents of the data packet to either compromise the destination node or to deceive it.

2.5.3.2 Accidental Altering

As the name suggests, accidental altering refers to the unintentional altering of the data packet during transmission. This type of altering happens as a consequence of other issues in the network and is most commonly associated with noise addition during transmission. It can also be caused due to other issues such as a data packet being misrouted or accidentally discarded by a device on the network. Such alterations to the data packet, although caused as a result of an accident, can still lead to dire consequences.

2.5.4 Confidentiality

The confidently of a network refers to the assurance provided by it that any given piece of data or any transmitted packet sent across the network is only accessible to the users it was meant for. The data packet should not be accessible to any other device on the network that does not have the authorization to view it. This is a major concern as, given the mesh network topology, each transmitted data packet travels through multiple user devices before reaching the destination. Hence

it is impossible to have the entire data packet confidential as a node requires certain information to be able to route the packet. The only possible way is to make the message part of the data packet confidential and keep any information required for the transmission of the packet in the header. A major threat to the confidentiality of a network is as below.

2.5.4.1 Man In The Middle Attack

As the name implies, a man in the middle attack involves a malicious node snooping in on an active connection between two nodes. The malicious node capturing the transmission data can not only listen in on the entire conversation but also modify the data transmitted causing issues. If a node is able to access multiple connections and can affect multiple nodes, the attack can cause enough discrepancy in the network to saturate it.

2.5.5 Non-Repudiation

The ability for a network to maintain a proper security depends on an important factor, the ability to assure the sender and receiver of a data packet. This allows for non-repudiation, that is, a node cannot send a packet to any given node and then deny sending it; similarly, a node cannot deny the receipt of a packet it has received. This feature comes in handy for detecting anomalies in the network by helping in locating and isolating the device responsible for it.

2.5.6 Anonymity

As user privacy is an important security concern for a network, any information provided by the user must be protected and used safely. This involves keeping any information that may aid in the process of user identification private and discrete.

Such information can include details about the device being used, contact information about the user, the location of the user and much more.

2.6 Key Security Aspects for Distributed Systems

The security provided by a network can usually be broken down into significant factors that collaborate and contribute to the overall strength of the network. Each of these factors are supported by certain key aspects of the techniques used and how well they are implemented. In this section, we will discuss the three major components that need to be present in any given security framework or model.

2.6.1 Authentication

Authentication is an important component of the defence mechanism against major security threats to the network as it prevents any unauthorized users from gaining access to the network directly [92, 93]. It also keeps a check on the authorized users to detect any anomalies caused as a result of an attack or if the device is compromised. The authentication also prevents impersonation attacks by verifying user and device details when a connection request is initiated.

Authentication in a traditional network is achieved through a designated authority that maintains a record of all the users and their corresponding details. These details are referenced to the details provided or collected during a connection request and based on certain algorithms the authenticity of the user is calculated. This process is however different in a wireless mesh network as it does not employ any centralization. Hence, in order to authenticate a device, the technique must be updated to adapt to the distributed control and ever-changing network topology.

Various schemes and models have been suggested to provide adequate authentication fit for a wireless mesh network, these can be combined in two types:

2.6.1.1 Centralized Authentication

This scheme is directly ported over from the traditional networks and employs a centrally situated entity to maintain a record of user details against which any authentication requests are judged. This is, however, a bit different from the traditional approach as it can also apply multiple entities in a large network to act as a central entity for a given fraction of the network. This scheme is used by the Identity-based Proxy Group Signature (IPGS) scheme [94], Ariadne Routing Protocol [95], Authenticated Routing for Ad hoc Networks (ARAN) [96] and many others. The authentication entities can be referred to as trusted parties, control nodes, verification gateways, certification authority etc. by different models. The functionality of these entities, however, remains the same.

They either request the user/device information or are provided with the same and are required to verify the details provided. The decision taken by an authentication authority decides if a device joins the network. The data collection for authentication can be done in two ways as well. A new device may be given network access prior to the authentication process so as to provide the authentication entity with the details required directly. This, however, raises certain concerns as a device is provided with access to the network without verifying its credentials and is hence less preferred. The more preferred option is to provide partial monitored access to the network using which the new device can provide the details required for the authentication process itself.

2.6.1.2 Decentralized Authentication

The decentralized authentication mechanism uses a comparatively less central entity reliant approach as it embraces the distributed nature of the mesh network. In this method, the data collection process is undertaken by edge devices

(authenticated devices that are already part of the network) and in some cases, they can even perform the authentication process. This is achieved by designing the authentication mechanism to rely more on challenges that only a verified user device can solve and have less reliance on user information verification.

The decentralized authentication can be achieved through a pre-set authentication challenge for all network devices or using a tree/parent-child approach. Such an approach is used in privacy-aware secure hybrid wireless mesh protocol (PA-SHWMP) [97], secure hybrid wireless mesh protocol (SHWMP) [98], Ticket-Based Authentication Mechanism [99], secure efficient distance vector routing (SEAD) [100].

The challenge based authentication works on the principle that any given user device has the ability to successfully solve a presented challenge only if it has been registered with the network. This is usually achieved by setting up the network to use public-private key pairs (Discussed in Section 2.6.2) and using said keys for authentication purposes as well. Due to the simplicity of the approach, any edge device on the network is able to both collect relevant data and authenticate the user device. In a tree or parent-child approach, a parent or branch node is considered the authentication authority and is able to authenticate a child node. Once authenticated, the child node becomes a parent node and repeats the same process to add any new node that wishes to connect to the network.

2.6.2 Encryption

The second defence against possible snooping or data tampering during transmission is achieved using end-to-end encryption. This enables the network to assure that the data sent over the network is only accessible to the source device and the destination device which may include multiple authorized devices. Encryption is achieved by using certain algorithms and keys to convert plain text into

non-readable scrambled text which can only be un-scrambled using the same algorithm and key used in the first place as shown in Figure 2.7. End-to-end encryption can be achieved in two ways, by using symmetric encryption or by using asymmetric encryption [101].

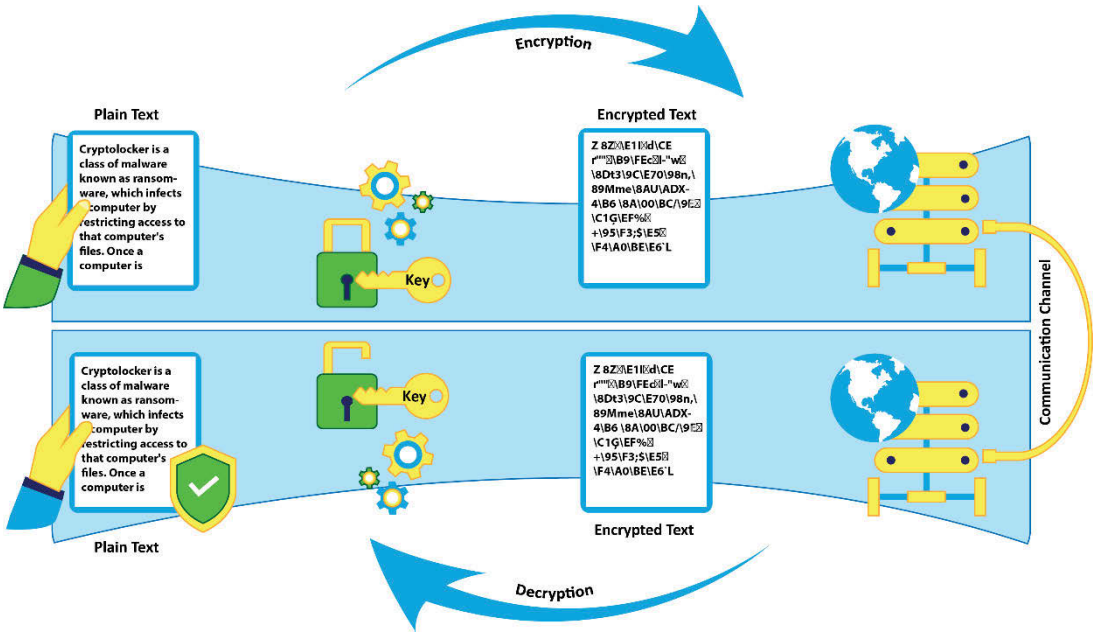

Figure 2.7 Encryption - Decryption Process

2.6.2.1 Symmetric Encryption

In symmetric encryption, a single key is used to encrypt the plaintext (Figure 2.8) using a symmetric algorithm such as AES [102-104], 3DES [104-106] and Blowfish [106, 107]. The key size used is directly proportional to the strength of the achieved encrypted text. As the key used for encryption is the same as the one used for decryption, hence the scheme is referred to as secret key encryption. The secret is generated using random or pseudo-random values and is usually of alphanumeric type. This secret key must be exchanged between both the devices that wish to communicate. The source node will use the key to encrypt the plaintext data it wishes to send to the destination. This encrypted data will then be placed in a packet

to be sent across the network securely and can only be decrypted at the destination node using the same secret key.

Although the encryption performance is faster as compared to the asymmetric technique, the symmetric algorithm does face some issues. If a single encryption key is used for the entire network, it means that anyone on the network can view the data. To mitigate this, each device must create a new secret key for each device it wishes to communicate with. This also adds the additional complexity to maintaining a record of all the secret keys and which device they correspond to.

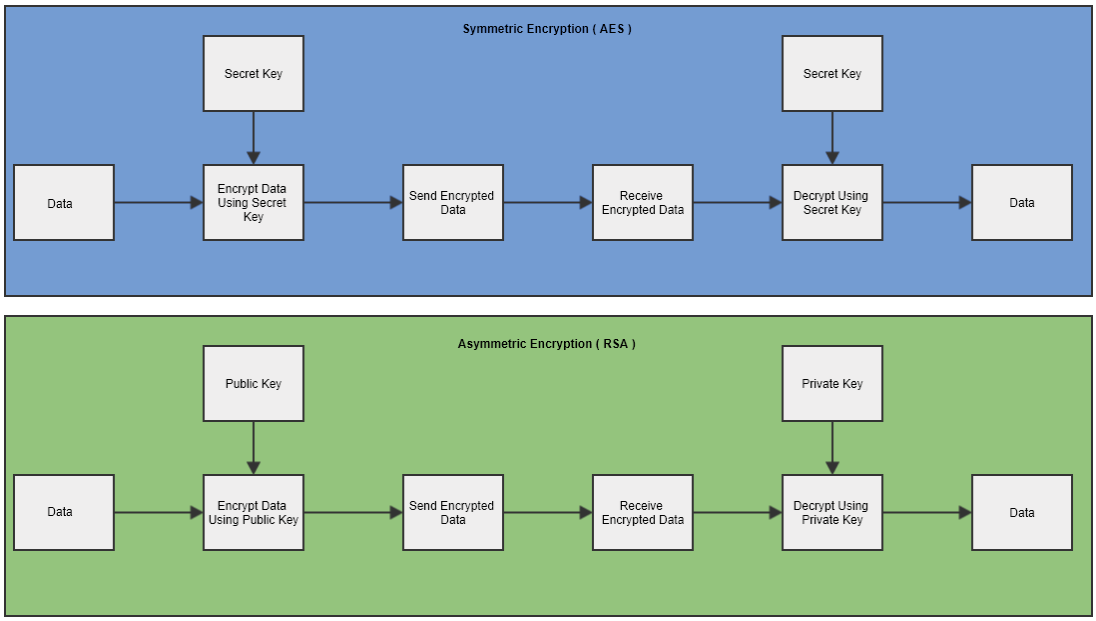

Figure 2.8 Symmetric and Asymmetric Encryption

2.6.2.2 Asymmetric Encryption

Unlike the symmetric encryption which uses the same key for both the encryption and decryption process, asymmetric encryption technique uses two keys as depicted in Figure 2.8. The keys used are always created in pairs, referred to as the public key and the private key, and so the scheme is also known as public-private key encryption. Some well-known examples of asymmetric encryption schemes are RSA [108-110] and ECDH [111, 112]. The keys are named according to their purpose

in the algorithm; the public can only be used to encrypt the data whereas the private key can only be used to decrypt the data. Having two keys to perform different tasks makes it easy for devices to maintain keys.

If two devices wish to communicate, they only need to share their public keys with each other. As the public key can only be used to encrypt the data, it can be released openly into the network for any node to use if it wishes to communicate. The private key on the other hand never leaves the host devices assuring the integrity of any data packet received. This method also uses fewer keys as each device does not require a separate set of keys for each possible communication. Each device must only maintain a set of public keys for the devices they wish to communicate with. This also ensures that is a device is compromised and a private key is lost, it will still not affect the rest of the network as the attacker will only be able to read any packets destined for the compromised node.

2.6.3 Key Management

As discussed in the in the previous sections, the network uses different types of crypto keys to perform encryption and can also be used for authentication purposes. Hence, the crypto keys play an important part in the security framework by strengthening other components of the framework. Key management techniques for any network can be divided into two main components, the key generation process and the key distribution process. These two processes work independently of each other but contribute equally to the strength of the network.

2.6.3.1 Key Generation

The key generation process is designated to create the crypto keys that will be used by the device for encryption and in certain cases authentication. The crypto keys can be generated using various techniques depending upon the requirement. As the crypto keys are of two types, symmetric (secret) or asymmetric (public-

private), the technique used must be selected accordingly. For each secret key or public-private key pair generated, there are certain factors that govern the strength provided by the keys when in operation. These factors include:

Seed Value

The seed value in an initial string of characters is used to create the key. The randomness of the characters in a seed value determines the strength of the key. This randomness can be true randomness captured from a source such as the clicks from a Geiger Counter [113], using the voice or thermal noise captured by sound equipment [114, 115] or using a light source [116]. The randomness can also be generated using pseudorandom sources that use certain algorithms to generate almost random numbers. Although true random sources and considered better than pseudorandom sources, the random sources have, at any given time, collected only a finite set of available characters as compared to the infinite number of characters that a pseudorandom generator can provide anytime.

Key Length

The length of the key is also considered important as it defines the number of characters available to the encryption mechanism to use for encryption. The more characters a key has, the greater its length and the greater the time it would take to guess the combination. This is because the number of possible combinations increases exponentially with the increase in a number of characters used as well as the type of characters used. For example, a 2-character binary based key will have 4 possible combinations as compared to a 10-character integer based key that can have 10,000,000,000 possible combinations, thus increasing the complexity and thereby utilizing more time to calculate.

2.6.3.2 Key Distribution

The key distribution process involves the exchange of newly generated crypto keys amongst devices and any possible central entities on the network. The key exchange is a crucial stage as any compromise or tampering with the keys can sabotage the other components of the security framework. The key exchange involves setting up a communication channel with a device or central entity in order to send the crypto key. The type of key being exchanged defines the type of security required for the transmission channel. A secret key requires a secure connection or a trusted party connection to be exchanged safely as it can be used for both the encryption and decryption process, whereas a public-private key pair can use an open channel as only the public key needs to be exchanged which can only be used to encrypt the data. Various existing schemes present different ways of achieving the same [112, 117-121] through the use of a certification authority responsible for storing and distributing keys, hierarchical based key distribution schemes, one-way tree based distribution, multi-key distribution and many more.

2.7 Summary

The wireless mesh network is already under constant improvements and at the same time faces new threats. As we presented in our discussions, there are various flaws in the current network models that are applied over the mesh networks as they impose the same structure followed by a traditional network. The mesh network topology being unique requires its own set of routing mechanism and protocols to help derive the functionality and expandability further than that of a traditional network. The new set of routing protocol must consider the important parameters that are of importance to the network and work towards optimizing individual components alongside strengthening the collaboration within them.

With the consideration of wireless mesh specific parameters, more features will be added to the network model to further expand its productivity. In addition to the change in the network model, the wireless mesh network will also be subject to new security threats that must also be addressed. Given the distributed topology, a mesh network has a much vivid set of security threats as compared to a traditional network, such as the worm-hole attack is only effective in the mesh network. The security framework must incorporate the crucial security aspects, as described in Section 2.6, to be able to provide enhanced levels of security within a wireless mesh network.

Chapter 3

Secure Geo-Location Oriented Routing Network Model

From this chapter onwards, we will begin to explore research problems along with proposed solutions for dynamic distributed routing. With any distributed system, a major concern arises with the routing protocol being used for connectivity or data transmission. The use of legacy/traditional protocols in distributed networks models, especially the mesh network, results in various limitations towards its actual implementation capabilities. In addition to the limited functionality, the security of distributed networks also requires attention as each data packet travels through multiple devices/nodes making it susceptible to vulnerabilities. In this chapter, we present our proposed Geo-Location Oriented Routing (GLOR) network model and an overview of the security framework it incorporates. The GLOR network model, unlike legacy systems, includes multiple new features which are no longer bound by the drawbacks of the legacy protocols and can be used to incorporate improved security measures.

3.1 Introduction

Mesh networks are known for their reliability as they are comprised of several devices (nodes) interconnected together to form a network. The messages are relayed using either a flooding technique or a routing technique. This is achieved by hopping the message from one node to another until it reaches the destination. The mesh network also has the ability of self-healing [122] allowing the network to operate when a node breaks down or when a connection becomes unreliable by automatically creating a new one.

Wireless mesh networks have been around for a while and have been used to build distributed networks, connect satellites for satellite calling and even for data collection over large areas. The mesh network topology has been under examination and experimentation to achieve a network model that is self-sustained, secure, scalable and dynamic. In the past few years, it has been identified that mesh networks hold the potential of becoming the network of the future, however, only a few attempts have been made to achieve it.

The current versions/implementations face various challenges, amongst which a major concern arises from the fact that the data packet is re-routed and hopped from one node to another [85]. This results in a limitation to the number of nodes a network can maintain. If the number of nodes increases beyond the threshold of the central controller/access point, it would be unable to manage the network. Such a scheme may encounter various issues resulting from the failure of the central controller/access point.

In pursuit of overcoming such limitations, a new network model needs to be developed. However, the design of the new network model incorporates several features that could not be achieved using current routing protocols, hence we developed the Geo Location Oriented Routing (GLOR) protocol. It is a secure, smart

and dynamic solution for the new mesh network model. Since devices are becoming smarter and possess higher hardware configurations, GLOR protocol incorporates various new features and a totally remodelled approach towards security, authentication, packet routing, network formation and addressing scheme.

In this chapter, Section 3.2 discusses existing approaches, routing models and traditional protocols for mesh routing. It also discusses the origin of these protocols, their implementation and limitations. Section 3.3 presents our proposed scheme Geo-Location Oriented Routing and discusses the new network model, smart packet design and a new addressing scheme. This section also discusses the functioning of the GLOR protocol, how it differentiates from traditional protocols and the security model being used. Section 3.4 presents our implementation and the results to validate our model. The chapter is finally summarized in Section 3.5.

3.2 Existing Approaches

There have been various proposed models and approaches to achieve a dynamic and self-sustained wireless network; however, to the best of our knowledge, only a few were ever implemented as detailed in Section 2.4. Below is a summary of some relevant implementations and their features.

The Smart Phone Ad-hoc Networks (SPAN) project [23, 64] that uses an updated version of OLSR (Optimized Link State Routing), showed promising capabilities for off-grid communication. However, it also highlighted a major flaw in the OLSR routing protocol which causes the network to self-saturate due to the use of excessive 'hello' packets while trying to build the neighbour table. Similarly, the Several Project [68] allows live voice calls using the mesh network. However, this project incorporates the Delay Tolerant Network scheme and in only supported a limited number of compatible devices. The approach also includes an external

hardware called the “Mesh Extender” which made the network dependent on the hardware and hence less stand-alone.

FireChat [73] uses a simple broadcast routing technique and severely lacks security as each message is sent out to every device on the network in hope of reaching the destination. The BRIAR Project [74] on the other hand is designed to provide secure and resilient peer-to-peer communication with no centralized servers and minimal reliance on external infrastructure. However, the approach once again follows Delay Tolerant Network (DTN) and to implement high levels of security, the devices do not communicate directly unless their owners share common contacts.

3.2.1 Routing Models

All the above network implementations are based on two major transmission techniques, namely Flooding/Broadcasting and Unicast/Multicast.

3.2.1.1 Flooding/Broadcast Technique

In the flooding/broadcast technique, each node in the network retransmits the received packet to all connected nodes thereby flooding the network in the hope that the packet would reach the destination node. This method was implemented by Open Garden [73] in their mobile application ‘FireChat’. This approach is applicable to a large network but it increases the load on each node as with the increase in the number of nodes, and with each node retransmitting every packet, the traffic on the network increases rapidly. This results in the use of more resources and in some scenarios, it could even lead a device to crash. In addition, the communication in the network is open and each packet of data is received and read by every other node in the network thereby compromising the privacy.

3.2.1.2 Unicast/Multicast Technique

The Unicast/Multicast technique is about implementing a predefined path through which a packet is sent. It supports a limited number of static devices/nodes as the network is converted into a map to calculate paths. Each node broadcasts a "HELLO" message and stores the location of every other node on the network for calculating a route when a packet is to be transmitted. This makes it difficult to upscale the network as the addition of devices makes it difficult to keep mapping the network and storing the information.

The routing protocol also has a major flaw; it was found to saturate the network with "HELLO" packets during normal operation as the number of connected devices increased. In order to support a large number of devices, it requires a central node/entity/gateway that stores all information regarding the nodes and controls the network by calculating routes which negate the very basic principle of a mesh network, its ability to be self-sustained.

3.2.2 Legacy Routing Protocols

Since the introduction of the Mesh Network, a few network protocols have been developed and various others have been modified to work with mesh topology [24, 26] as discussed in Section 2.3. Some relevant examples include the Optimized Link State Routing (OLSR) protocol [36, 48, 66, 123] developed using optimization of the classical link state algorithm and modified in accordance with the requirements of a mobile wireless LAN. The key concept used in the protocol is Multi-Point-Relays (MPRs) [46].However as depicted by the SPAN project [64], the protocol is known to flood the network with “Hello” messages resulting in network saturation.

Ad-hoc On-Demand Distance Vector (AODV) routing [49] offers quick adaptation to dynamic link conditions, low processing and memory overhead, low network utilization, and determines unicast routes to destinations within the ad hoc

network. The protocol performs well in small networks, but as the number of nodes increases, it starts to fail. Zone Routing Protocol (ZRP) [58], and Bordercast Routing Protocol (BRP) [57] are hybrid routing framework based on various routing protocols. Each node maintains a route within a local region (known as the routing zone). Knowledge of the routing zone topology is used by the protocol to improve the efficiency of the routing mechanism. As ZRP/BRP is a combination of various other protocols, it also inherits both the merits and demerits of other protocols.

3.3 Proposed Geo-Location Oriented Routing Protocol

Geo Location Oriented Routing (GLOR) is designed as a hybrid routing protocol with the aim of supporting large, dense and dynamic networks without compromising the reliability and security of the network and the devices within it. To achieve this, a new network model was created that is unlike any legacy or AD-HOC model. A distinguishing factor of the new approach is that unlike existing approaches, it utilizes the high-performance capabilities of current smart devices which possess better hardware configuration. The smart approach provides a new platform for improvements in various aspects as discussed below.

Reverse Network Model: In our approach, the nodes are responsible for maintaining the network as compared to the traditional networks where the nodes are maintained by the network. For example, the node address (geo-location) is calculated and provided by the node itself instead of the network providing one to it. Similarly, other tasks like the node registration process, node monitoring, packet routing and address allocation are also monitored by the nodes themselves.

Security Framework: The routing protocol uses a combination of security measures including hybrid encryption, hybrid authentication, monitoring and a new

key management scheme. Each of these components has been discussed further in Section 3.3.1.

New Addressing Scheme: Unlike traditional methods, the GLOR approach uses geo-location of a device as its address (described in Section 3.3.2). The geo-location is obtained using GPS or is calculated by nearby nodes minimizing the need for a central control entity. This provides us with the instantaneous position of each node, like dots on a fixed canvas, to be used for data transmission.

Smart Packets: As the protocol uses geo-location for node addressing, the data packet format is modified to include location information and associated parameters. Once a packet is created, a predefined route is not required as all the necessary information required for routing is contained within the packet data. The packet knows the destination address, i.e. the geo-location of the destination node as well as its current geo-location. From this information, the packet automatically calculates its transmission path (described in Section 3.3.4)

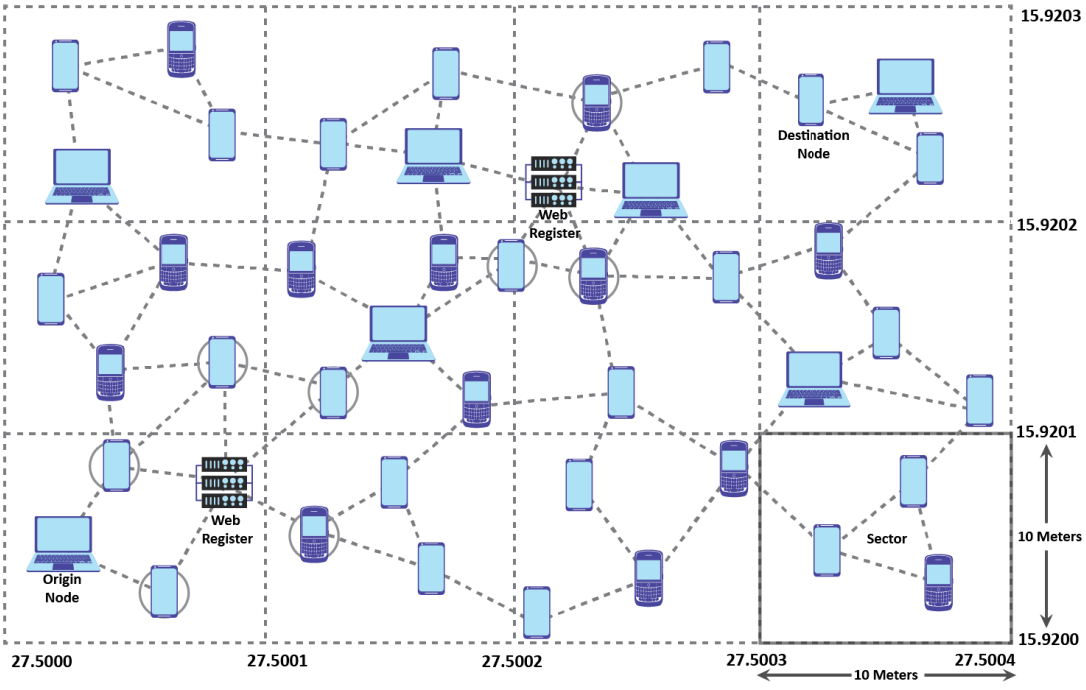

Figure 3.1 Network Scenario

The protocol function is further explained using a network scenario as shown in Figure 3.1. The various steps involved in the routing process are shown in Figure 3.2 along with the line of connectivity.

As shown in Figure 3.1, the node is an electronic device (e.g. Smart-Phone, Laptop, and Tablet) that implements Geo-Location Oriented Routing (GLOR). It can be classified as a normal node or a web node depending on their connectivity. A normal node has the capability to connect to other devices wirelessly given they implement the GLOR protocol, the web node is a normal node with the additional capability to connect directly to the Web Register. A node X is said to be the neighbour node of Y if there exists a link between the node X and node Y.

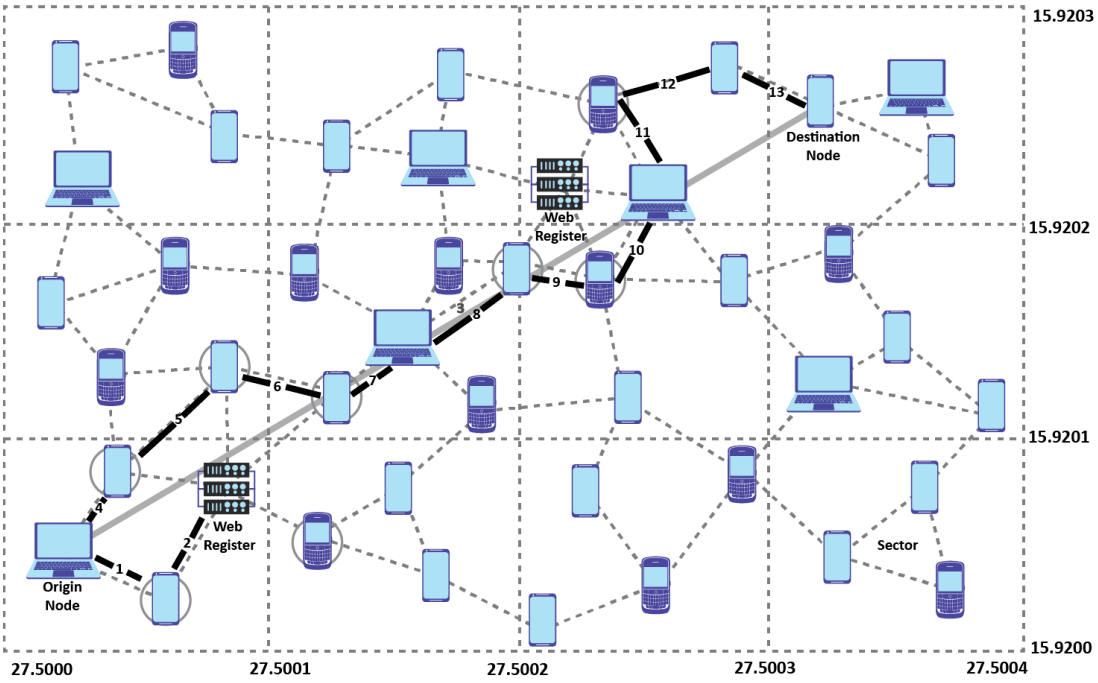

Figure 3.2 Different Steps of Routing Process

The nodes can be identified using two factors, the node address and the unique ID. The node address is the Geo-Location of the Node, i.e. its latitude and longitude measured up to 4 decimal places and the node's unique ID is a one-time generated Unique Identification number assigned to the Node alongside its MAC address during

its first registration on the network. The web register is a cloud-based database dedicated for storing vital information about nodes, including their MAC address, unique ID, address, and current state. The Sector for a Node can be defined as a group of its neighbouring nodes. This helps improve the accuracy as each node in a sector knows other nodes in that sector.

Mobility Model: The GLOR protocol follows the random mobility model to address the movement of physical devices. For experimental validation, the devices are deployed according to the random waypoint mobility model [124] (described further in Section 3.3.5).

3.3.1 Security Framework

A security model comprised of multiple components is used in the Secure-GLOR protocol. It is implemented through different network levels, each focusing on an important aspect of routing. These aspects are hybrid encryption, hybrid authentication, monitoring and key management. Each of these is summarized below and are explained further in the following chapters.

3.3.1.1 Hybrid Authentication

The authentication starts during the Node Registration process (described in Section 3.3.3) and is a vital part of the network model. Once a new device connects to the network, its neighbour node/s (which has/have been previously authenticated) collect the device data as mentioned in the node registration process. If more than one node can communicate with the new device, both nodes compare the collected data to improve the authentication process.

Once the web register confirms that the device is new to the network, the user must manually enter their personal details including the selection of the unique ID, which doubles as their contact number, and the generation of a public-private key

combination to be used for encryption and decryption purposes. These details (excluding the private key) are sent to the neighbouring node which is then encrypted and sent to the web register for safekeeping and referencing.

As soon as the new device clears the authentication process, it's status is changed to authenticated node. From this point onwards all the data sent to and from the device is encrypted using the public-private key combination created earlier. The hybrid authentication process used in the Secure-GLOR has been modified to address the various scenarios a device may face while in a distributed environment. These scenarios have been discussed and implemented in Chapter 4.

3.3.1.2 Hybrid Encryption

The encryption method plays a major role once the device/node has successfully authenticated itself. Each node has its own unique pair of public and private key-pair out of which only the public key is stored on the web register to be used to communicate with the node. This ensures that each packet sent over the network can only be decrypted by the node it was destined for.

The data packets are encrypted using a session key, which is exchanged by using the public key of the destination node, at the origin node, where the public key is obtained from the web register (described in Section 3.3.4). The packet also contains the public key of the origin node so that the destination node can similarly encrypt any reply with the provided key.

The end-to-end encryption makes the network very secure as only two nodes can see the contents of the packet; the origin and destination. Any node, that a packet encounters during transmission, can only read the header containing the packet information but cannot access the message/data it contains. This also prevents any unauthorized nodes trying to access the data or impersonate an authenticated node. The proposed hybrid encryption technique has been further

expanded, analysed and tested on the Secure-GLOR network model, more in-depth details regarding which have been presented in Chapter 5.

3.3.1.3 Key Management

As the Secure-GLOR uses cryptographic keys in various instances, it is important to manage such keys properly. In addition, as both authentication and encryption rely on these keys to provide security in the network, it is also important to keep them secure. Even so, there are various threats that may interfere with this process and in doing so compromise the security of the network.

The Secure-GLOR network model uses the Multi-Path Anonymous Randomized Key (MPARK) exchange technique to minimize any interference or intrusions during the key exchange[125]. The MPARK technique and its working are discussed further in Chapter 6.

3.3.1.4 Monitoring

The monitoring of the network is conducted through the web register by observing the timely updates it gets from the nodes in the network during the node update (described in section 3.3.4). For instance, the web register uses the geo-location data of the node to determine if a node is trying to impersonate another node by comparing their location and the displacement between the updates. It is an important aspect that none of the node's identity or its location is spoofed, hence our proposed protocol is designed to guard against such malicious attacks through the monitoring aspect.

It checks if a node's unique ID is showing two different geo-locations at the same time or is switching locations at a pace that is physically impossible. If either is the case, the nodes are flagged. This data is then used to find nearby nodes to check which of the flagged nodes is real and which one is trying to impersonate. The data

collected is compared and processed to find which of the flagged nodes are real and accordingly the impersonator is blocked and reported.

The monitoring can also help in finding lost/stolen devices as once powered on they will connect to the network and can be easily tracked using their location. In addition, its neighbour devices can once again aid in confirming its identity and help the appropriate authorities to confiscate the item.

3.3.2 Node Addressing

The GLOR protocol uses IPv6 addressing format for storing the geo-location as it makes the new network model compatible with traditional networks which support IPv6. The IPv6 protocol offers 32 hexadecimal bits, which are further divided into eight groups of 4 hexadecimal bits each. The first 4 groups are used for storing the node location and the last 4 groups store the sector and cluster information of the node.

The first 4 groups are sub-divided into 2 groups to store the latitude and longitude information corresponding to the node's geo-location. The first digit represents whether the value of latitude is positive (denoted by 1) or negative (denoted by 0), while the following 3 digits are the number before the decimal point.

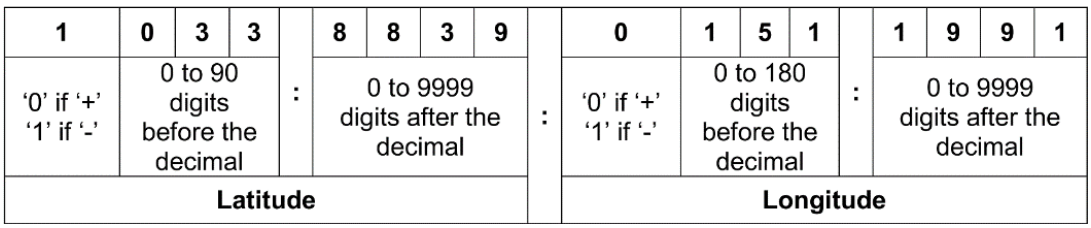

Figure 3.3 Addressing Scheme (Part 1)

The next 4 digits are the number after the decimal point. The longitude is represented similarly. The first 8 hexadecimal bits denote the latitude and the next

8 bits denote the longitude, both with an accuracy of 10 meters. Figure 3.3 shows the structure used to store latitude and longitude.

The next 4 groups store the cluster number and the sector number. Each sector represents 100 square meters of land and is defined using the latitude-longitude system. For example, the area enclosed by latitude 1.0000 to 1.0001 & longitude 1.0000 to 1.0001 represents a sector as depicted in Figure 3.1. The cluster is a combination of predefined sectors. Figure 3.4 shows the sector-cluster structure used.

0	0	0	1	:	0	0	1	2	:	3	4	5	6	:	7	8	9	0
Cluster					Sector													

Figure 3.4 Addressing Scheme (Part 2)

The sectors and clusters are calculated automatically based on the latitude and longitude of the node, which is based on the international standard representation of geographical point location by coordinates (ISO 6709). A sample scenario is shown in Fig 3.5.

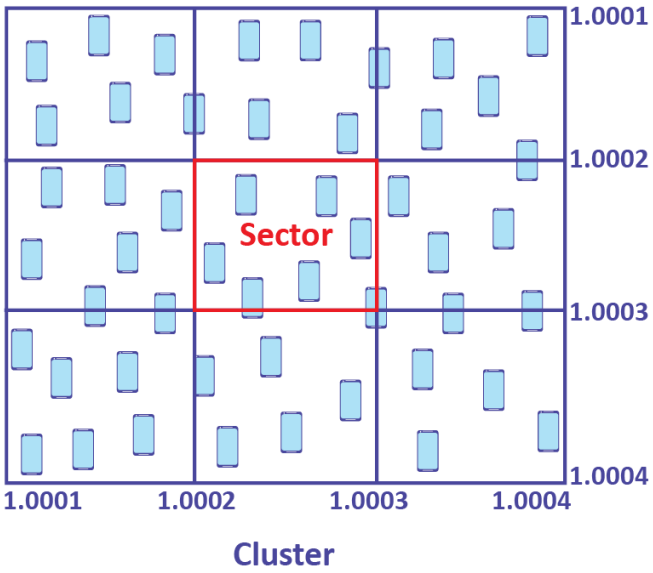

Figure 3.5 Sector and Cluster Formation in the Network

3.3.3 Node Registration

The node registration process is initiated when a new device requests to connect or an existing device re-connects to the network. Once powered on, the device scans its surroundings for neighbouring nodes to initiate the connection request through. Once the neighbour list is populated, it selects the closest neighbour node (implementing GLOR protocol) and sends a 'Hello' message to initiate the handshake. On completion of the handshake the new node requests neighbour node to start its registration process.

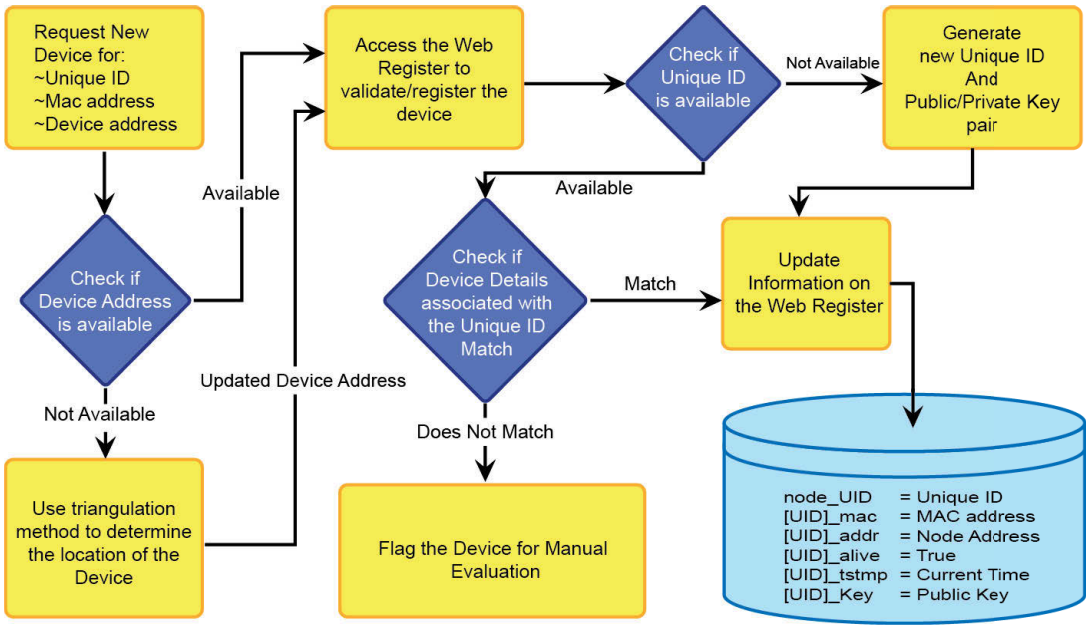

Figure 3.6 Node Registration Process

The process, as shown in Figure 3.6, includes details about the device/user information which is collected, how it is validated and the steps involved in the registration process. The first registration for any node is manual as it requires the user to fill in details manually in order to complete the registration process. If a device is re-connection to the network, it does not have to re-register itself. The new

device must, however, pass the authentication by solving a challenge created by the web register (by encrypting it) using the public key of the new device.

3.3.3.1 Web Register

As discussed earlier in the chapter, the web register is a cloud-based dedicated database used to store device information. It can be accessed by any authenticated node that has access to the internet, or through a neighbour node which possesses internet access. The web register acts as the yellow pages of the network and improves the performance and accuracy of the network.

Web register, being a key element of the network, is not a central or control node. The network can function without its presence by following a Sector-Broadcast Progression. According to this method, the origin node sends out packets aimed in the direction of its four neighbouring sectors. As each node keeps a record of all the devices in their sector, it can check if the destination node exists in the sector. If yes, then the packet is relayed to it, if not then the packet is forwarded to the neighbouring sector. In comparison to the simple broadcast method, the sector-broadcast helps to lower the load on the network.

3.3.4 Smart Packets

The GLOR protocol defines the functioning of a node in the network. This includes the universal specifications of GLOR messages, Node Registration, Packet Format & Transmission, Neighbour discovery and Routing.

3.3.4.1 Packet Format

GLOR protocol communicates using a modified packet format. The purpose is to keep it simple to reduce the load on the network. It helps incorporate different types of information in a single transmission which optimizes the use of max frame

size. The basic layout of the packet has been updated to include the new addressing scheme and is represented in Figure 3.7.

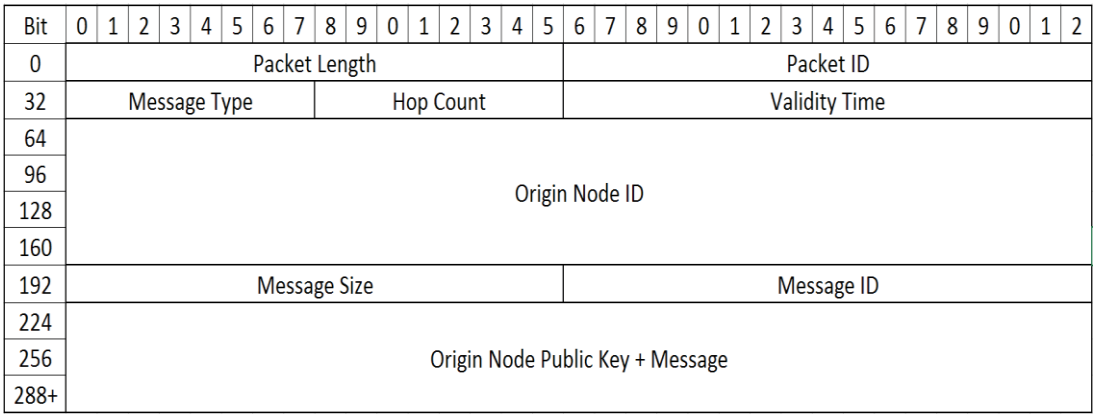

Figure 3.7 Packet Format (Omitting TCP/IP Headers)

The simple design and minimized header size help the packets carry more data and reduce overhead. Various components of the packet are described below:

- Packet Length - It is the length of the packet (in bytes).
- Packet ID - The Packet ID or PID is an identifier and must be incremented by one each time a new GLOR packet is transmitted
- Message Type - It indicates the type of the message that is being transmitted.
- Hop Count - It is the number of hops a message has attained. It is incremented every time the packet is retransmitted.
- Validity Time - It is the maximum time during which the information of the packet is considered valid. If a node receives a packet with Validity Time = 0, the packet is discarded.
- Origin Node ID - This is the ID of the node that originally generated the packet. It is not to be confused with the Source Node ID in the IP header as the Source ID is updated each time to the address of the intermediate node whereas the Origin Node ID remains constant.

- Message Size - It is the total size in bytes measured from the beginning of “Message Type” till the end of the message.
- Message ID - A unique ID is provided to each message by the Origin Node. It is incremented by one for each message. As a message can be divided into multiple packets, Message ID helps in identifying the separately received packets and grouping them accordingly.
- Origin Node Public Key - It is the public key of the origin node that is to be used by the destination node for encrypting any data it wishes to send back.
- Message - It is the actual data being sent to the destination node.

3.3.4.2 Packet Formation

This process defines how a packet is generated. Once the origin node is ready to send a packet, it requests the address of the destination node by providing the destination node’s unique id to the web node. The web node initiates a request to accesses the web register to retrieve the information represented as step 1 and 2 in Figure 3.2. Once it gains access to the web register, it checks if the unique id exists in the registry. If the unique id is not linked to a node, a ‘not_found’ response is then sent to the origin node.

If the unique id is found, the next step is to check if the node is still connected to the network or not. This is done by accessing the destination nodes [UID]_alive parameter. If the destination node is still connected to the network, the origin node receives the destination node’s address. However, if the destination node is currently offline, the origin node receives a ‘not_alive’ message.

Once the Origin Node receives the destination node address and the public key, it creates the packet with the appropriate information and encrypts the message part (which also includes its own public key to ensure any reply will be encrypted as well) using the destination node’s public key. The packet is then processed according to the type of the message defined in the next section.

3.3.4.3 Next Hop Calculation

Once the origin node has gathered the required information and the packet is created, the next hop is calculated according to the method depicted in Figure 3.8. The same procedure is also used at every hop for the calculation of the next hop. The various parameters and math involved in the calculation are as follows.

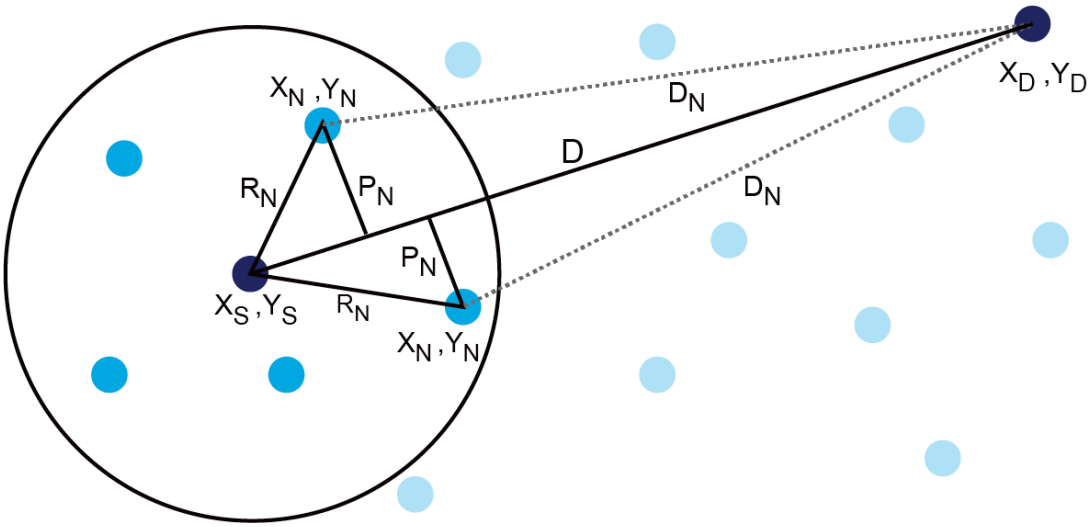

Figure 3.8 Next hop calculation

Table 3.1 Components for Next-Hop Calculation

Variable	Description
X_S, Y_S	Geo-location of the source node
X_D, Y_D	Geo-location of the destination node
X_N, Y_N	Geo-location of the neighbouring node/s
R_N	The distance of neighbour node from the source node
D_N	The distance of neighbour node from the destination node
D	The distance of source node from the destination node
Line D	A straight line from the source node to the destination node
P_N	The distance of neighbour node from line D

The distance P_N is calculated using the following equation:

$$P_N = \frac{|(X_D - X_S)(Y_S - Y_N) - (X_S - X_N)(Y_D - Y_S)|}{\sqrt{(X_D - X_S)^2 + (Y_D - Y_S)^2}} \qquad (1)$$

The neighbour node is selected using the geo-location of the source node and the destination node. Using these two location details as two points on a graph, a straight line is plotted and then the neighbour node closest to the line and farthest from the source node is selected and the packet is transmitted to it as shown in Figure 3.8. A neighbour node can be selected as the next hop if it satisfies the following conditions:

- The node should be alive and in the neighbour table of the source node
- The node's distance from the destination node (D_N) should be less than or equal to the distance from the source node to the destination node (D).
- If there are two or more nodes that satisfy the above conditions, then a node is given preference based on the following.
  — Its distance from source node (R_N) is greater
  — Its distance from destination node (D_N) is less
  — Its distance from line D (P_N) is less

If two or more nodes satisfy the conditions, the one with less load is selected. This process repeats itself until the packet reaches its destination.

3.3.4.4 Packet Processing and Forwarding:

Once a node receives a packet, it examines the header and its contents based on the message type. The process that takes place once a packet is received is presented in Figure 3.9. In order to make the system robust, the origin node can transmit the same packet multiple times or to multiple nodes. This can result in each node receiving the same packet multiple times.

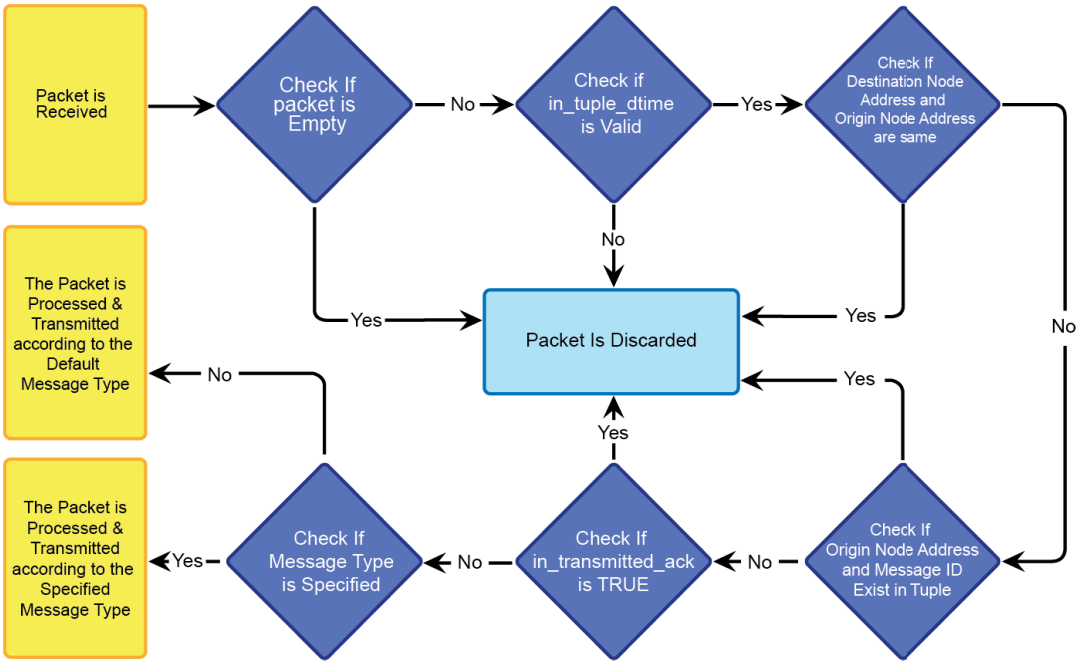

Figure 3.9 Packet Processing and Forwarding

To prevent retransmission of the same packet, each node creates a duplicate tuple with details about the packet (in_origin_address, in_message_id, in_transmitted, in_transmitted_ack, in_tuple_dtime). In_origin_address is the origin node address, in_message_id is the message id of the packet, in_transmitted is a Boolean which represents if the packet was transmitted further or not, in_transmitted_ack is also a Boolean representing if the acknowledgment was received or not after the packet was transmitted, in_tuple_dtime is the time after which the data expires and the tuple will be discarded.

3.3.4.5 Default Packet Forwarding

Once a packet has been processed, it is checked if it has been transmitted before. If so, the acknowledgement is checked. If an acknowledgement has been received, the packet is discarded; otherwise, the packet details are updated and it is transmitted, as shown in Figure 3.10.

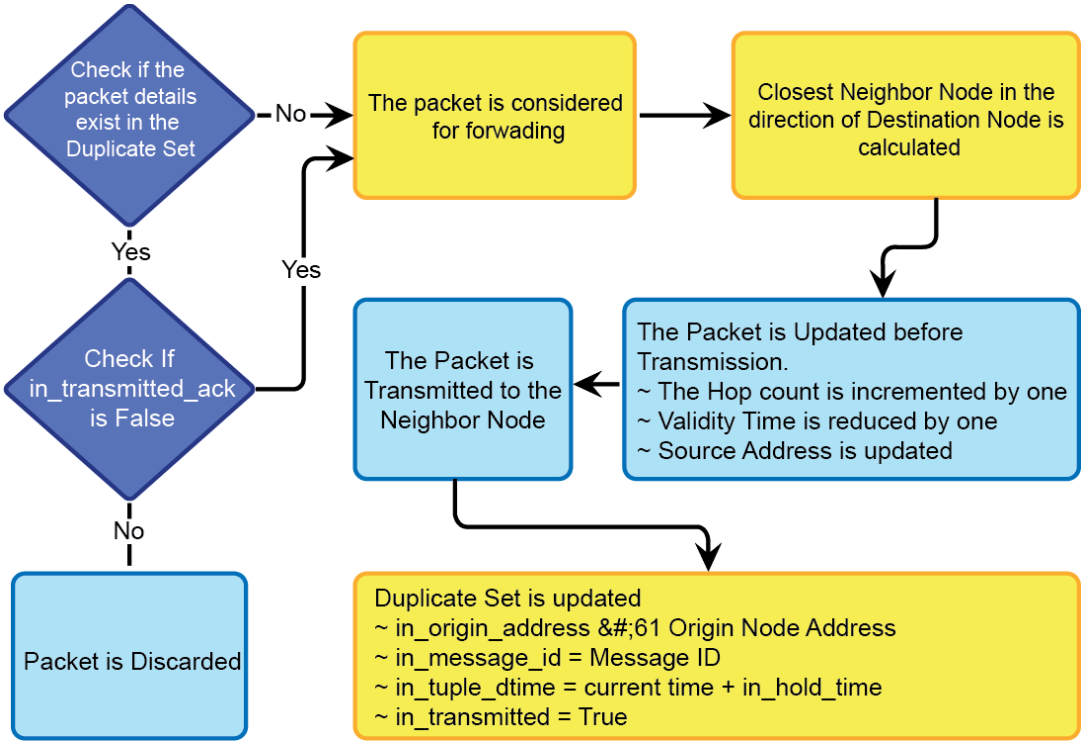

Figure 3.10 Default Packet Forwarding

3.3.4.6 Node Updating

Each node connected to the network will send an update to the web register informing it about its location change or to acknowledge that it is still connected to the network. The 'node_check' process handles this task and is repeated at regular intervals. The process first checks if the node has changed its location; if yes, then it checks if the change in location is more than 10 meters. If the change in location is more than 10 meters, the web register is sent a request to update the location and [UID]_alive status. If the location change is less than 10 meters, only a [UID]_alive message is sent.

3.3.5 Random Waypoint Mobility Model

The random waypoint mobility model [62] follows a standard mobility mode where a device's location is changed every 60 seconds (referred to as the pause

time). A device starts of at a given point on the 2D plane (x_i, y_i) and once the pause time is reached, it will randomly select another set of coordinates on the 2D plane (x_{i+1}, y_{i+1}) in a random direction (d_{i+1}). Once the new coordinates are selected, the device moves there with a random speed (s_{i+1}) defined by min-speed and max-speed. The coordinates are calculated based on the equation described below:

$$x_{i+1} = x_i + s_i \cos d_i \quad (2)$$

$$y_{i+1} = y_i + s_i \cos d_i \quad (3)$$

The direction d_{i+1} and the speed s_{i+1} are calculated as below:

$$s_{i+1} = \alpha s_i + (1-\alpha)\bar{s} + \sqrt{(1-\alpha^2)s_{x_i}} \quad (4)$$

$$d_{i+1} = \alpha d_i + (1-\alpha)\bar{d} + \sqrt{(1-\alpha^2)d_{x_i}} \quad (5)$$

In the equations, i is defined as the time interval; α is the tuning parameter defined as $0 \leq \alpha \leq 1$ (where 0 reflects total randomness and 1 represents linear motion), it is used to vary the randomness; $\bar{s}$ and $\bar{d}$ represent the mean value of speed and direction; s_{x_i} and d_{x_i} are random variables from Gaussian distribution.

3.4 Results and Validation

The structure of GLOR protocol vastly differs from the legacy protocols due to its unique working parameters and design. The proposed routing protocol has been developed using C# in Visual Studio Enterprise 2015 IDE and the required basic libraries were created from scratch for the new network model. The machine used for simulation is an Alienware 13 powered by a 6th Gen. Intel i7 (3.1 GHz.) CPU and 16GB DDR3L RAM.

3.4.1 Environment Setup

The devices/nodes are represented using objects available in the Visual Studio 2015 IDE, and each object (referred to as device or node) runs GLOR protocol by default. The geo-location is calculated using the X-Y coordinates of the node according to its placement on the 2D plane.

The devices are randomly distributed over the plane during testing. The web register is designed using a localized database that stores the node information. Other components such as the data packet design and various variables being used in the routing process are also defined in the library files.

- The testbed has been created for simulation with the following assumptions:
- The nodes are uniformly distributed across the plane.
- The nodes have already been authenticated and have a unique id.
- None of the nodes fail during the operation.
- All nodes have the capability to calculate their location.
- No packet is dropped during the transmission process.
- Each node has a direct/indirect connection to the web register.

3.4.2 Simulation and Observation

Once the simulation starts, the nodes first calculate their geo-location (in this case it's X-Y coordinates on the plane). The next step is to search and connect with neighbouring nodes. This step also involves creating the neighbour table which helps with the next-hop selection during the routing.

In addition to the above steps, each node also updates its information on the web register. Once the devices have connected and the network is formed, two random devices are manually selected to start an exchange of a predefined send-

acknowledge packet. The scenario also traces the path taken by the packets as shown in Figure 3.11. The preliminary test conducted using 20 nodes provided vital first-hand information about the setup. It also helped update the next-hop calculation method as it was found to go into an infinite loop in some scenarios.

Once the appropriate modifications were made, the final tests were conducted with 72 nodes. The test showed promising results as the GLOR protocol was able to route packets through multiple devices. It was also observed that different packets from the same device may take a different route based on its calculation of the next hop and the availability of neighbouring nodes.

3.4.3 Results and Discussion

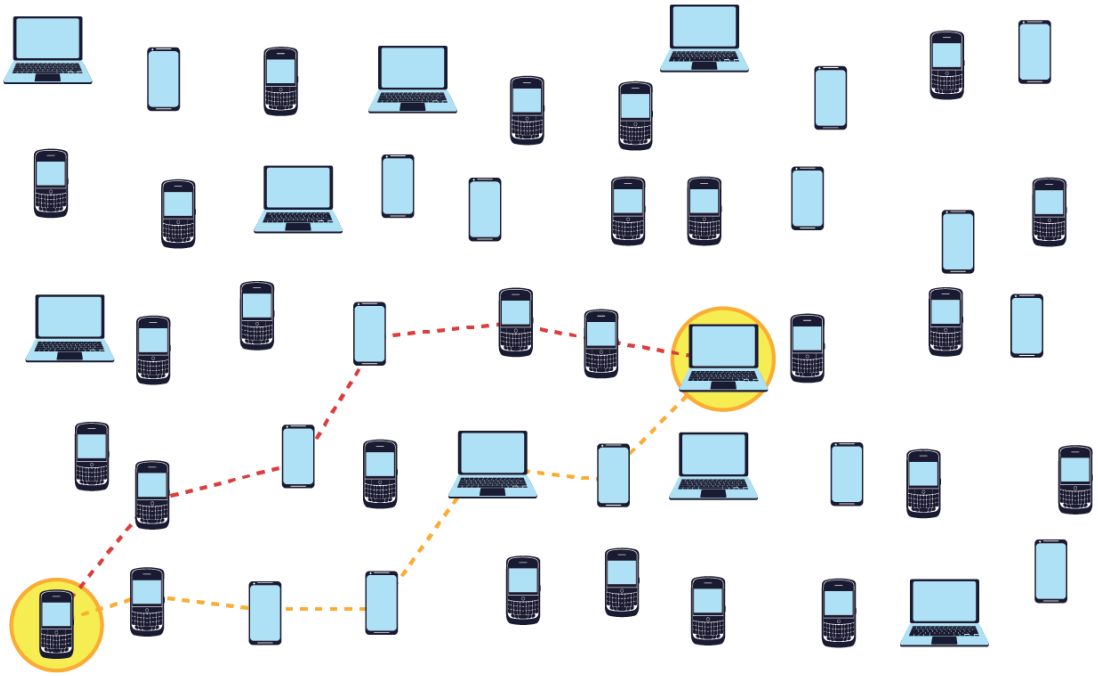

Figure 3.11 Instance Showing Packet Route Trace

As shown in Figure 3.11, the acknowledgement packet from the destination node (depicted with a light dotted line) did not take the same path as the original packet (depicted with a dark dotted line). Current analysis proves that the routing

can easily and efficiently adapt to a dynamic mesh network. The graph in Figure 3.12 shows the time taken for a complete sent-acknowledge cycle.

In addition to the time, the simulation result also shows that hardware utilization for a node to forward a packet is 11% of CPU usage with 5MB of additional RAM for a duration of 10 milliseconds on an average.

The results obtained from the simulation further help to improve the performance and reveal various scenarios which might not have been addressed so far. As the simulation moves forward, it is expanded to include more nodes and observe the behaviour and performance in such scenarios.

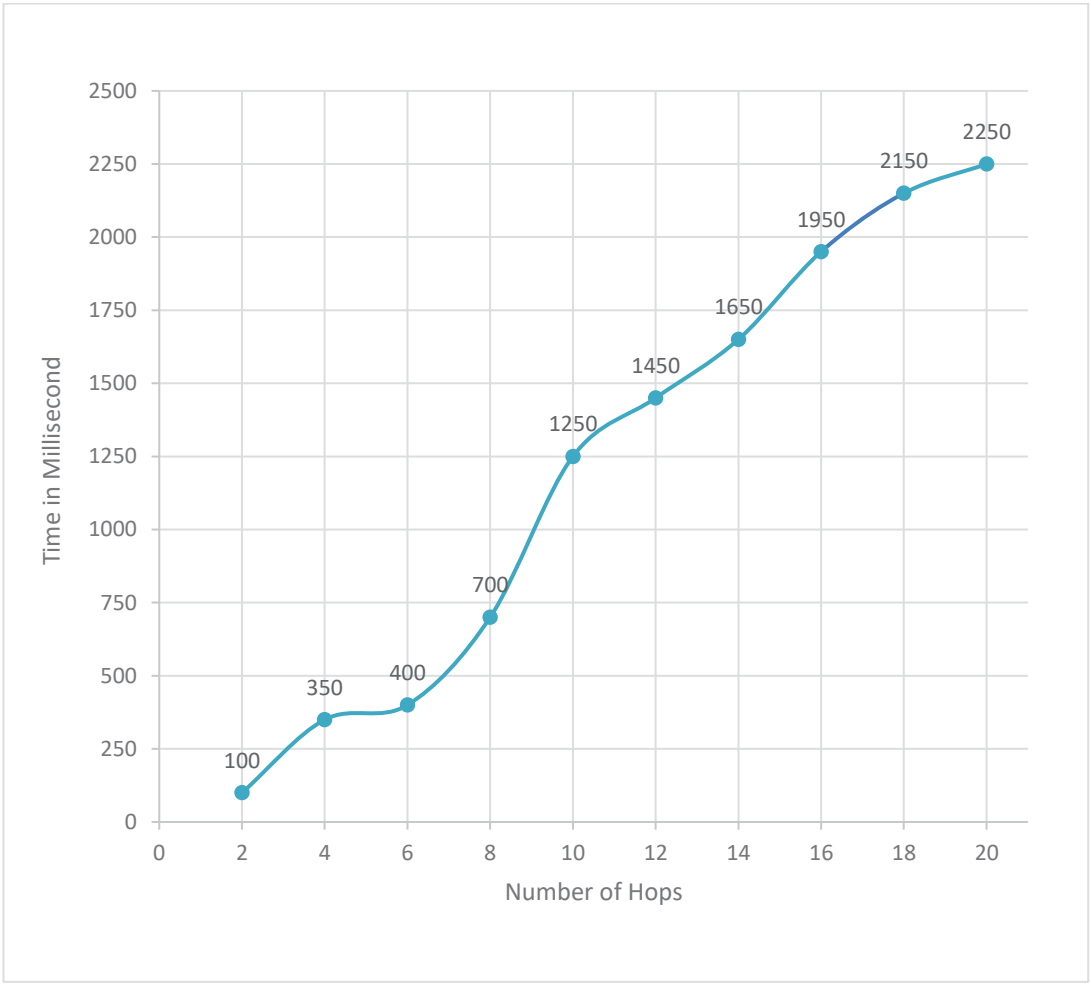

Figure 3.12 Message Roundtrip Time (+/- 10%)

Since the first testing, GLOR protocol has been under constant modification to increase the efficiency, enable better security and be able to handle unique/exceptional scenarios that might arise in real-world scenarios such as the dead loop or the “V” tip [126, 127]. Further development of the GLOR protocol will enable it to identify such exceptional scenarios and take appropriate measures to avoid them or find a way around them.

3.5 Summary

The new network model along with a new decentralized addressing scheme, the GLOR protocol for data transmission and network connectivity together with the security framework comprising of hybrid encryption, hybrid authentication, MPARK management together form a very robust communication network providing end-to-end security. The innovative model also provides a new platform for further development of this routing technique as the many new features presented can be used to improve upon both the performance and security of the network.

As it is not governed by the limitations of legacy protocols, the GLOR network model also opens the doors for the development of various applications that can perform considerably better as compared to the legacy network model. In the chapters ahead, we will discuss more about the security framework in detail and test its performance on the proposed Secure-GLOR network model.

Chapter 4

Authentication Mechanism for Distributed Networks

Authentication is an essential part of any network and plays a pivotal role in ensuring the security of a network by preventing unauthorised devices/users from accessing the network. As dynamic wireless mesh networks are evolving and being accepted in various fields, there is a strong need to improve the security of the network. The mesh network's features like self-sustainability and self-healing make it a great network but these features are undermined when rigid authentication schemes are used. This chapter proposes a hybrid authentication scheme for such dynamic mesh networks under three specified scenarios; full authentication, quick authentication and new node authentication. The proposed schemes are applied to our dynamic mesh routing protocol, Secure - Geo Location Oriented Routing Protocol (Secure - GLOR). Simulation results show our proposed scheme is efficient in terms of resource utilization as well as defending against security threats.

4.1 Introduction

The mesh networks have evolved very considerably in the past few years and are being used extensively for the device to device communication. They feature a self-sustained network model where the data is transmitted from one point to another by the concept of hopping. This is achieved by connecting multiple devices together and then sending the data from the host device to the next device and repeating this process multiple times until the data finally reaches the destination node. This can be achieved through unicast/multicast routing where a single path or multiple paths are used to send data or by flooding the whole network with the data.

A typical mesh network can be either static or dynamic, depending upon the type of connected devices. If stationary/fixed devices form the mesh network, it is known as a static mesh network. It can be wired, wireless or a combination of both depending upon how devices connect to each other. The mesh network comprises various noble features such as self-configuration, which allows the devices to connect and create the network without any external control entity. It involves low operating costs as the network is composed of user devices, which are easily set up by implementing an identical protocol on all devices. The maintenance of the network can be considered by the device owners while providing robustness as multiple devices create redundant connections. A dynamic size can adapt according to the number of devices. In addition, the self-healing properties also make wireless mesh networks an ideal network choice for the future.

However, it is important to note that a mesh network sometimes is unable to perform at its full potential as the current/legacy protocols limit the extent of its features and size [85]. Aspects such as IP addressing requires a central server to manage the network which makes the network dependent on the server destroying its self-configuration properties [18].

As the mesh network works by sending data through multiple devices, these devices have access to the data flowing through the network [19]. This raises various security concerns as the network becomes prone to even the simplest attacks such as eavesdropping which can compromise the privacy of the users and the integrity of the network.

Hence, along with various other network models, security has also become a must for mesh networks too. Recently, various security models have been developed for the mesh network [18, 19, 36, 45, 48, 49, 75, 82, 85, 96, 100, 102, 103, 108, 123, 128-133]; however, the security models themselves have become a factor which prevents the mesh from expanding. To provide high levels of security, a central controller is used to manage the network, preventing the network from expanding and working at its full potential.

The Chapter begins with a summary of related/existing security schemes, how they implement authentication and their limitations in Section 4.2.1. Section 4.2.2 defines the problem statement whilst providing a summary of the main challenges. Following that, Section 4.3 presents our proposed authentication scheme with various scenarios and how they work to provide better security. Section 4.4 presents the simulation results and analysis and the chapter is summarized in Section 4.5.

4.2 Current Approaches and Problem Statement

The wireless mesh network is prone to various types of threats ranging from basic attacks like Denial of Service, Eavesdropping, Spoofing and Flooding all the way to more advanced attacks such as the Sinkhole attack, Impersonation, Sybil attack, data redirect, and many more [18, 19, 36, 48, 49, 66, 85, 102, 103, 108, 123, 128, 132]. In essence, most of the attacks in mesh networks can be traced to a compromised device or an unauthorised access to the network. Hence,

authentication plays a crucial and integral part in preserving the security of the network by keeping the attackers away from accessing the network.

4.2.1 Current Approaches

The wireless mesh network has some well-known routing techniques such as the OLSR (Optimized Link State Routing) [36, 48, 66] and AODV (Ad hoc On-Demand Distance Vector) [49] as discussed in Section 2.3; both these schemes have almost no security aspect by themselves, however, there have been modified versions that include security. SOLSR is a secure version of the OLSR protocol which uses features like message authentication codes (MAC's), timestamping and cryptographic signatures to prevent the most common attacks on OSLR such as identity spoofing, link spoofing, tc packet spoofing [45].

Similarly, SAODV is a secure version of AODV protocol which implements two mechanisms, digital signatures [128] and hash chains, to provide security and ensure the integrity of the network [82]. There are various other protocols such as ARAN (Authenticate Routing for Ad hoc Networks), which uses a single trusted key pair for the whole network to ensure security [96]. SRP (Secure Routing Protocol) [75], SMT (Secure Message Transmission Protocol) [131] and SAR (Security-Aware Ad Hoc Routing Protocol) [133] use a shared secret key amongst devices to verify packets. Protocols like SEAD (Secure Efficient Ad Hoc Distance Vector Routing Protocol) [100] and SLSP (Secure Link State Routing Protocol) [130] use a table-driven approach along with time-synchronization or secret key exchange and other similar featured protocols.

However, most security schemes are either based on flooding technique, which increases the network load on each device, or they require an existing security association between the devices. Others such as OLSR are known to self-saturate the network just by overcrowding of Hello messages.

Hybrid Authentication is a must for multi-hop networks as it can provide redundant ways in which a device can authenticate itself or other devices [88, 134]. It is also certain that there is a need for an authentication server to verify and keep a check on all the authentications.

At the same time, there must exist other equally secure ways of authentication so that the network can function even if the authentication server is unreachable [135, 136]. A similar approach that implements hybrid authentication is presented in [129] which discusses a multi-level model for authentication. However, the model can only be applied to static wireless mesh networks and not the dynamic wireless mesh networks.

4.2.2 Problem Statement

The dynamic wireless mesh network requires a dynamic security model comprising a new authentication scheme, which can adapt to various scenarios and still be able to provide high levels of security. As it is made up of mobile devices, which keep switching connections as they move, a static authentication scheme with rigid rules will slow down the network.

Given the topology, any given device trying to access the network may not be able to reach the authentication server due to several reasons, preventing it from gaining access to the network. This can result in an abundance of failed authentication requests and would rapidly slow down the network's expansion, lowering its coverage.

4.3 Authentication Mechanism

The GLOR model presents a basic authentication scheme [19], which is dependent on the web register for verification of the device details. However, getting access to the web register might not always be possible. This can result in a long delay for the new device to gain access to the network. In addition, the authentication process requires the devices to first establish a connection to the network and to be authenticated which poses a security threat to the network itself.

In order to make the authentication process faster and much more secure, we propose three scenarios which encapsulate all possible conditions a device can encounter while establishing a connection to the network. During authentication, the new device is kept in a sandbox, which prevents the new device from discovering any further details about the network. The new device is not provided network access until the authentication is successful. The three distinct scenarios are described as following.

Full Authentication: In this scenario, a device is reconnecting to the network and is authenticated by a node which has a direct or indirect link to the web register. On successful authentication, the network device will grant the new device network access along with the right to authenticate other devices.

Table 4.1 List of Components

Term	Component	Description
Node	Network Device	A device with an established connection to the network and is authorized to authenticate other devices.
Device	New Device	A device which wishes to join the network.
WR	Web Register	A database that stores network device information such as Unique ID, MAC, Address, Public Key and much more.

UID	Unique ID	A unique identifier generated and provided to each node by the Web Register. It is linked with each device's MAC.
ADDR	Geo-Location Address	Physical position (two dimensional) of the device determined through its latitude and longitude coordinates
K_{PU}	Public Key	RSA-2048 based encryption key pair used for authentication and End-to-End encryption. Each device gets its own key pair.
K_{PI}	Private Key	
K_{CR}	Crypto Key	AES-256 based encryption key provided to each device at registration.

Quick Authentication: In this scenario, the device is reconnecting to the network and is authenticated by a network device which does not possess a direct or indirect link to the web register at the moment. In this scenario, the network device itself carries out the authentication. On successful authentication, the new device is granted network access but not the right to authenticate new devices until the network device has verified the new device's information with the web register.

New Node Authentication: In this scenario, an unregistered device (which has never connected to the network) wants to join the network. For this scenario, it is vital that the network device maintains a direct or indirect access to the web register. This is required as all the device information collected must be recorded at the web register for pre-registration authentication and the registration process.

Once the Node has collected enough information about the Device, it decides upon the authentication scenario to be used. The decision on which scenario the device must pass through is based on the availability of the new device's unique ID and access to the web register as shown in Figure 4.1. The presence of UID implies that the new device has been registered and is re-connecting to the network. Table 4.1 lists various components of the hybrid authentication model and associated terms used to represent them.

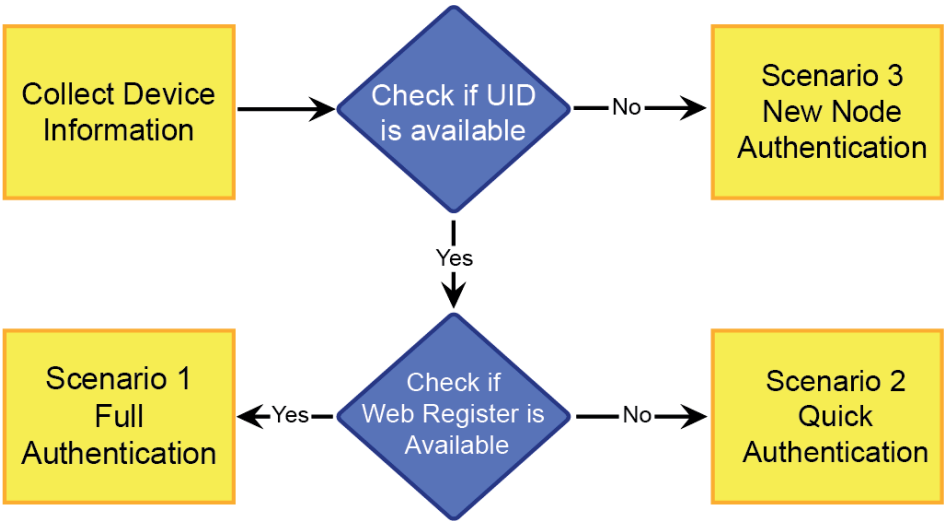

Figure 4.1 Authentication Scenario Selection

The authentication scheme is based on a challenge-response technique and uses a mathematical equation along with the encryption keys to verify the device. All the encryption keys that are used during the authentication process are stored in a TPM (Trusted Platform Module) style device. Such device is then used to prevent any unauthorized access to the sensitive information if a device on the network is internally compromised. The authentication scenarios are discussed in detail below.

4.3.1 Full Authentication (Scenario 1)

The steps in the full authentication process are divided into four major parts: Handshake, Device Information Collection, Challenge and Decision as shown in Figure 4.2. Individual processes are defined as follows.

4.3.1.1 Handshake

The very first step for the Device is to scan its surroundings for devices using the GLOR protocol. Once a Node (a device implementing the GLOR protocol and being connected to the network) is found, the Device will initiate a handshake request.

The Node will then respond to the request to complete the handshake. Once the Handshake is over, the Device requests the Node for network access, which then initiates the authentication process.

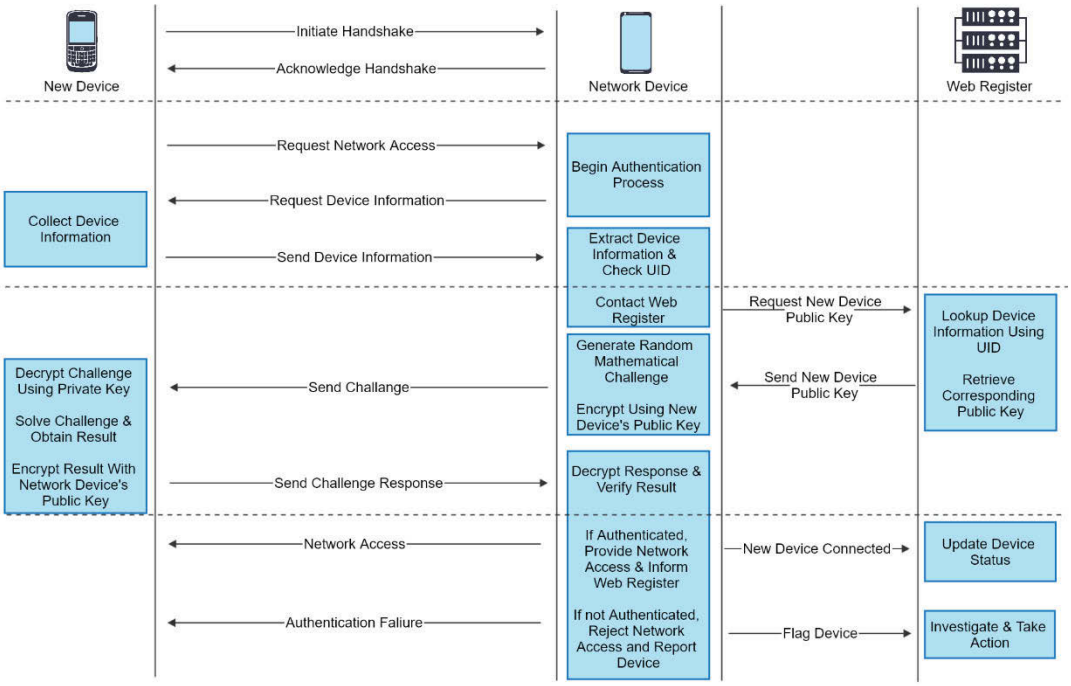

Figure 4.2 Full Authentication Process

4.3.1.2 Device Information Collection

Before the authentication process begins, the Node must first request the Device for its information including details such as UID, MAC, ADDR, etc. The Device must then provide the above-mentioned information to the Node as these details play an important role in verifying the status of the device.

The Node will first check if the Device has a UID as it is only provided to registered devices. Once the presence of UID has been verified, the device information is forwarded to WR.

Algorithm 4.1 provides details on the creation and the process of challenge-response used in scenario 1.

Algorithm 4.1 Scenario-1 Challenge

K_{PI}(D) - private key of D; K_{PU}(D) - public key of D
VAR - Variable; OPR - Operator; RLT() – Result;
CLN - Challenge; RES – Response

1. Get device encryption key

Node requests WR for K_{PU}(Device)

Node (*Device(UID||MAC)*) → WR

If WR found Device in the register and verified

WR → Node: (K_{PU}*(Device)*)

2. Create challenge

Node uses the random function to generate an equation

Node(Random) = VAR_1, VAR_2 & OPR_1

Node checks if the equation is valid

RLT(Node) = VAR_1 OPR VAR_2

If RLT(Computable) = True, Go to Step 3.

If RLT(Computable) = False, Repeat 2.

3. Send challenge

Node uses K_{PU}(Device) to encrypt challenge and add K_{PU}(Node)

CLN = K_{PU}(Device)[VAR_1 OPR VAR_2 || K_{PU}(Node)]

Node → Device: (CLN)

4. Solve response

Device uses K_{PI}(Device) to decrypt and solve challenge

K_{PI}(Device)[CLN] = VAR_1 OPR VAR_2 || K_{PU}(Node)

RLT = VAR_1 OPR VAR_2

Device uses K_{PU}(Node) to send the response

RES = K_{PU}(Node)[RLT]

Device → Node: (RES)

5. Verify response

Node extracts the response using K_{PI}(Node) K_{PI}(Node)[RES] = RLT(Device) If RLT(Node) == RLT(Device), Grant Net Access & Authentication Rights Node (*Device(Connected)*) → WR If RLT(Node) =! RLT(Device), Authentication Fail Node (*Device(Flagged)*) → WR

4.3.1.3 Challenge

Once WR receives the Device's information, it looks for the device records in its own database by referring to the UID. Once the details are found, they are compared with the Device's details provided by the Node. If the details match, the Device is verified and web register sends the K_{PU}(Device) to the Node.

Upon receiving the K_{PU}(Device), the Node will then create a random mathematical challenge where both the values and the operation will be chosen at random (e.g. "10 ^ 4", "74 / 3 * 4", etc.). This challenge will then be encrypted using the K_{PU}(Device) and sent across to the Device ensuring that only the device that possesses the K_{PI}(Device) (Stored in the Trusted Platform Module) will be able to decrypt the challenge and solve it.

To ensure there is no intrusion during the process, the Node will also send along its own K_{PU}(Node) so that the challenge response is also encrypted. The Device can now use K_{PI}(Device) to decrypt the challenge, solve the equation and use the K_{PU}(Node) to encrypt the result and send the response back.

4.3.1.4 Decision

Upon receiving the response from the Device, the Node will decrypt the response with K_{PI}(Node) and check the result. Once the result is verified, the Node will finally provide network access to the Device along with the right to authenticate other devices on the behalf of the network. The Node will also send an update to the

WR informing that the Device has gone through the authentication process and has been verified and provided network access.

The WR will update the ADDR and last seen information in its records for the Device and enable the right to authenticate. This will ensure no node can add another Device until it has been verified by the WR.

4.3.2 Quick Authentication (Scenario 2)

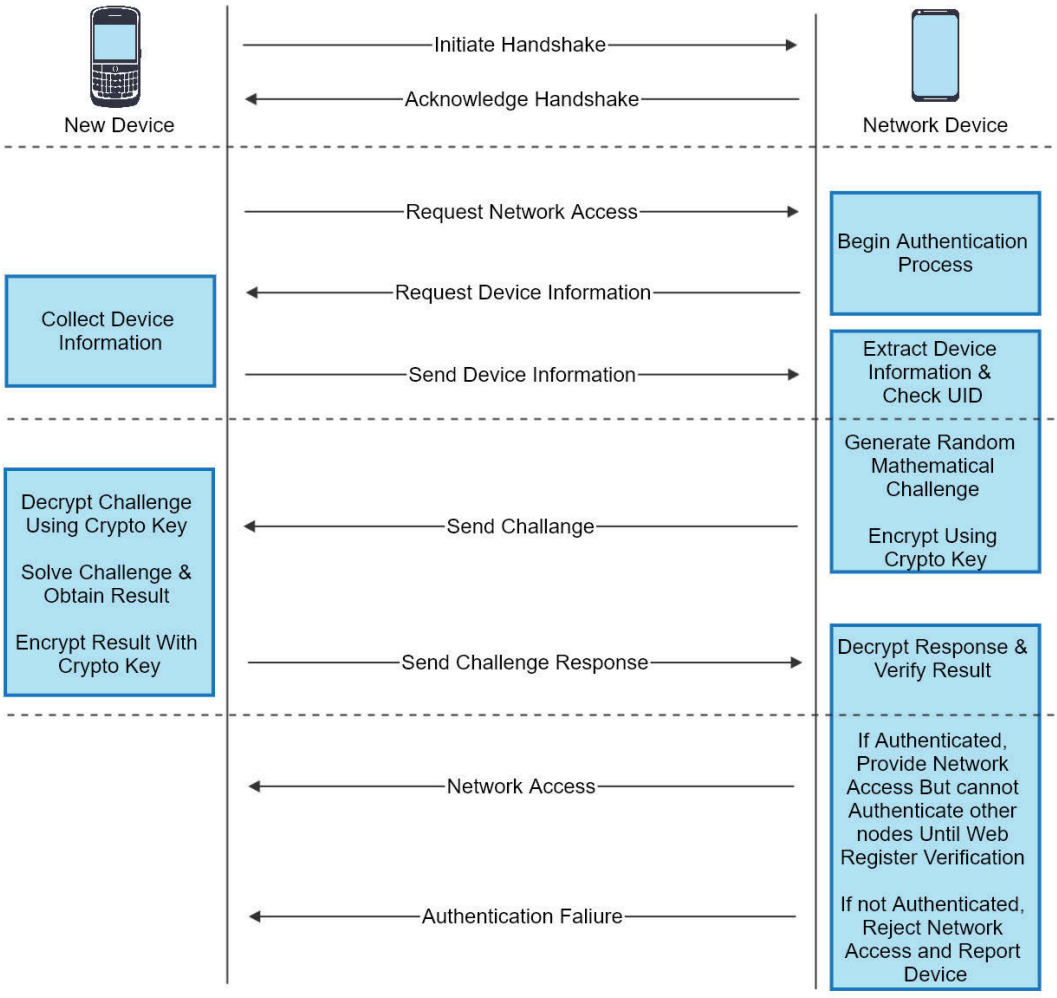

Figure 4.3 Quick Authentication Process

Like the full authentication process, the quick authentication process is also divided into four major parts: Handshake, Device Information Collection, Challenge and Decision as shown in Figure 4.3.

4.3.2.1 Handshake

This process is identical to the one used in the previous scenario.

4.3.2.2 Device Information Collection

Before the authentication process begins, the Node must first request the Device for its information which, includes details such as UID, MAC, ADDR, etc. The Device must then provide the above-mentioned information to the Node as these details play an important role in verifying the device.

The Node will first check if the Device has a UID as it is only provided to registered devices. Once the presence of UID has been verified, the device checks if it can access the WR.

Algorithm 4.2 presents the technical exchange that takes place during this authentication process.

Algorithm 4.2 Scenario-2 Challenge

K_{CR} - crypto key; VAR - Variable; OPR - Operator; RLT() - Result; CLN - Challenge; RES - Response

1. Create challenge

Node uses the random function to generate an equation

$$Node(Random) = VAR_1, VAR_2 \ \& \ OPR_1$$

Node checks if the equation is valid

$$RLT(Node) = VAR_1 \ OPR \ VAR_2$$

If RLT(Computable) = True, Go to Step 2.

If RLT(Computable) = False, Repeat 1.

2. Send challenge

Node uses K_{CR} to encrypt the challenge

$$CLN = K_{CR}[VAR_1 \ OPR \ VAR_2]$$

$$Node \rightarrow Device: (CLN)$$

3. Solve response

The device uses K_{CR} to decrypt and solve the challenge

$$K_{CR}[CLN] = VAR_1\ OPR\ VAR_2$$

$$RLT = VAR_1\ OPR\ VAR_2$$

The device uses K_{CR} to send the response

$$RES = K_{CR}[RLT]$$

$$Device \rightarrow Node: (RES)$$

4. Verify response

Node extracts the response using K_{CR}

$$K_{CR}[RES] = RLT(Device)$$

If RLT(Node) == RLT(Device), Grant Net Access

Wait for Connection → WR

Node (*Device(UID||MAC||Connected)*) → WR

If RLT(Node) =! RLT(Device), Authentication Fail

Wait for Connection → WR

Node (*Device(UID||MAC||Flagged)*) → WR

5. Create challenge

Node uses the random function to generate an equation

$$Node(Random) = VAR_1, VAR_2\ \&\ OPR_1$$

Node checks if the equation is valid

$$RLT(Node) = VAR_1\ OPR\ VAR_2$$

If RLT(Computable) = True, Go to Step 2.

If RLT(Computable) = False, Repeat 1.

6. Send challenge

Node uses K_{CR} to encrypt the challenge

$$CLN = K_{CR}[VAR_1\ OPR\ VAR_2]$$

$$Node \rightarrow Device: (CLN)$$

7. Solve response

The device uses K_{CR} to decrypt and solve the challenge

$$K_{CR}[CLN] = VAR_1\ OPR\ VAR_2$$

$$RLT = VAR_1\ OPR\ VAR_2$$

The device uses K_{CR} to send the response

$$RES = K_{CR}[RLT]$$

$$Device \rightarrow Node: (RES)$$

8. Verify response

Node extracts the response using K_{CR}

$$K_{CR}[RES] = RLT(Device)$$

If RLT(Node) == RLT(Device), Grant Net Access

Wait for Connection → WR

Node (*Device(UID||MAC||Connected)*) → WR

If RLT(Node) =! RLT(Device), Authentication Fail

Wait for Connection → WR

Node (*Device(UID||MAC||Flagged)*) → WR

4.3.2.3 Challenge

As the WR is not available or times out, the Node must follow the quick authentication process. As the Node cannot receive the K_{PU}(Device) from the WR, it uses the GLOR K_{CR} (a symmetric encryption key).

The Node will create a random mathematical challenge where both the values and the operation will be chosen at random (e.g. “10 ^ 4”, “74 / 3 * 4”, etc.). This challenge will then be encrypted using the K_{CR} and sent across to the Device, ensuring that once again only a registered device will be able to decrypt the challenge. This is possible because the K_{CR} is only provided to registered devices during their first registration and is stored in a Trusted Platform Module (which is known to be extremely secure) only to be accessed by the GLOR protocol for encryption and decryption purposes.

4.3.2.4 Decision

Upon receiving the response from the Device, the Node will decrypt the response with the K_{CR} and check the result. Once the result is verified, the Node will finally provide network access to the Device. However, the Node will not provide the right to authenticate other devices until a verification is done by the WR. The Node will now wait for an access to the WR and inform it once the connection is achieved and the Device is verified and connected.

The WR will check the device information against its records and if verified, it will provide the Device with the right to authenticate other Devices on the behalf of the network. The WR will also update the ADDR and last seen information in its records. This scenario introduces a new K_{CR} (AES 256) [102, 103] which, provides an alternate method for authentication if the WR is not available. The K_{CR} referred to here is universal and is saved inside a trusted platform module (or a trusted execution environment for devices that do not possess the hardware). The K_{CR} can

only be accessed by the GLOR protocol for encryption-decryption purposes in case no immediate access to the WR is available.

The Device can now use its K_{CR} to decrypt the challenge and solve it. Once solved the Device will again use K_{CR} to encrypt the result and send the response back to the Node.

4.3.3 New Node Authentication (Scenario 3)

In this scenario, we take into account the device that is connecting to the network for the first time; hence, it does not have any UID. In addition, the WR will not also contain any record matching the Device's information. Hence, a new record will be created as shown in Figure 4.4.

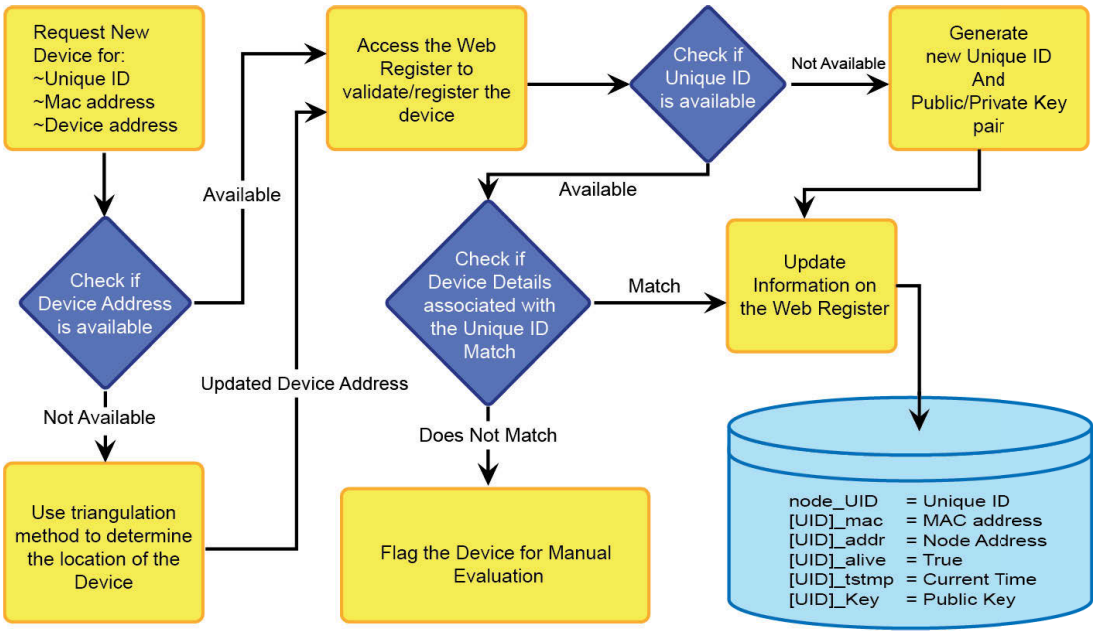

Figure 4.4 Registration and New Node Authentication

This scenario also incorporates the device registration process as defined by GLOR [18]. The new node authentication scenario is divided into four parts: Handshake, Device Information Collection, Verification and Registration.

4.3.3.1 Handshake

This process is identical to the one used in the previous scenario.

4.3.3.2 Device Information Collection

Similar to the previous scenarios, the Node first requests the Device for its information. The Device must provide the required information, however, unlike the first two scenarios, it would not contain any UID. On verifying that the Device does not possess a UID, the Node must begin the registration process on its own.

4.3.3.3 Verification and Registration

Before the Device can register, the Node must set up a secure connection to the Device as well as the WR to verify the details provided. To do so, the Device is asked to generate a new key pair K_{PI}(Device) and K_{PU}(Device), from which the K_{PI}(Device) is submitted to the trusted module and the K_{PU}(Device) is shared with the Node. Once the communication is secured, the Node will send the data to the WR for verification.

The WR upon receiving the Device's information will check if any matching records exist to make sure duplicate records are not found. If no duplicate records are found, the WR will create a record for the Device and generate a UID to map the device's information. The WR will then send the registration details to the Node, which will pass it on to the Device.

Once this process is complete, the Device will be provided network access by the Node and given the right to authenticate other devices on behalf of the network.

4.4 Results and Validation

The simulation for the authentication model using GLOR protocol has been developed in Visual Studio using C#. The machine used for simulation is powered by a 6th Gen. Intel i7 (3.1 GHz) CPU and with 16GB DDR3L RAM running Windows 10.

4.4.1 Environment Setup

The environment consists of two Smart Devices (both implementing the GLOR protocol), one of which being part of the network (Node) and the other attempts to connect to the network (Device). The Web Register (WR) is implemented using a local SQL database. The Device and Node have been allocated a maximum transmission speed of 11Mbps, which is an average speed of transmission based on the oldest non-legacy hardware still in use (Wi-Fi or Bluetooth). The transmission and processing times are calculated based on the processing power and transmission speed of the devices.

For the simulation environment, we consider following assumptions:

- None of the devices fail during the operation
- Both devices have the capability to calculate its Geo-Location (ADDR)
- There is no data loss during transmission.
- For scenario 1, the Node has a direct connection to the WR

4.4.2 Results and Analysis

The simulation involves the Device starting the authentication process by initiating the handshake with the Node. The simulation then proceeds along as defined in the scenarios. The simulation does not involve Scenario 3 (New Node

Authentication) as it is an extension of full authentication and hence, would have similar results.

Simulation is conducted individually for each scenario and used to collect information on the total transmission time of the data packets along with the average CPU and memory utilisation of the end devices involved. This provides us with valuable information about how the network performs under different conditions.

The simulation for Scenario 1 is conducted based on the model description from Section 4.3.1. The simulation starts with the devices authentication process. We then capture the time taken for the authentication process to complete. Figure 4.5 displays a timeline of the authentication process starting at '0' seconds and finishing at '3.3' seconds while mapping the key tasks in between.

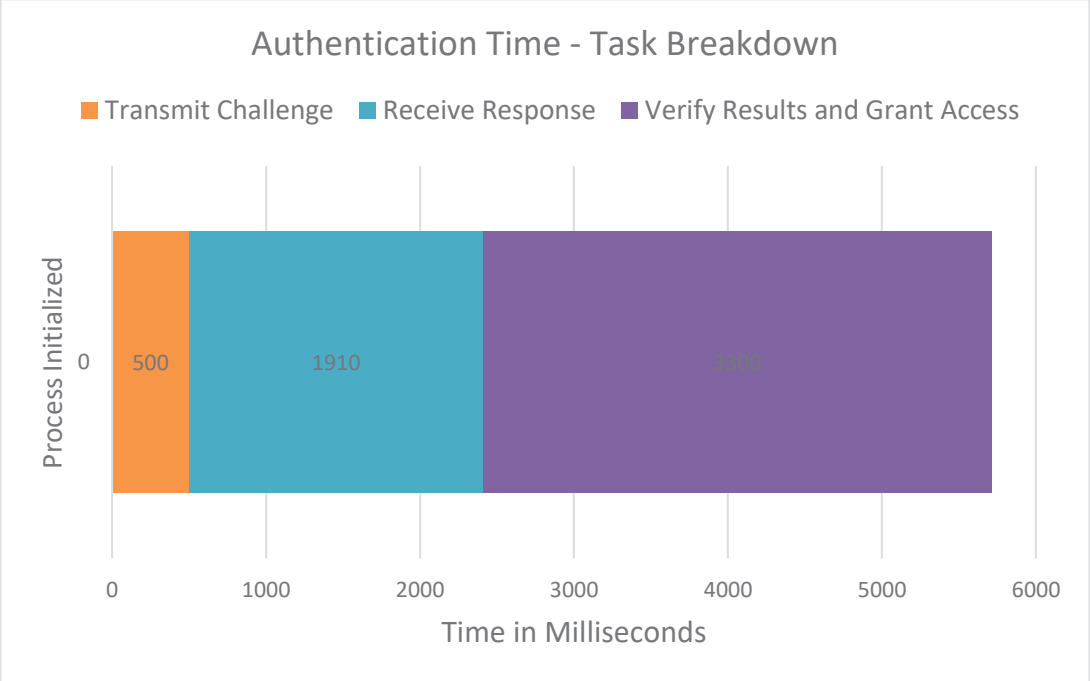

Figure 4.5 Scenario 1 Timeline

The authentication process begins once the handshake is completed and is denoted by '0' on the time scale in Figure 4.5 and Figure 4.6. Once the node has created the challenge it sends it to the device, the time taken until this point is calculated and presented in the figure.

The next key task is calculated when the device receives the response and addresses it. Finally, the authentication process ends with the node verifying the response received from the device and deciding whether to provide access or not.

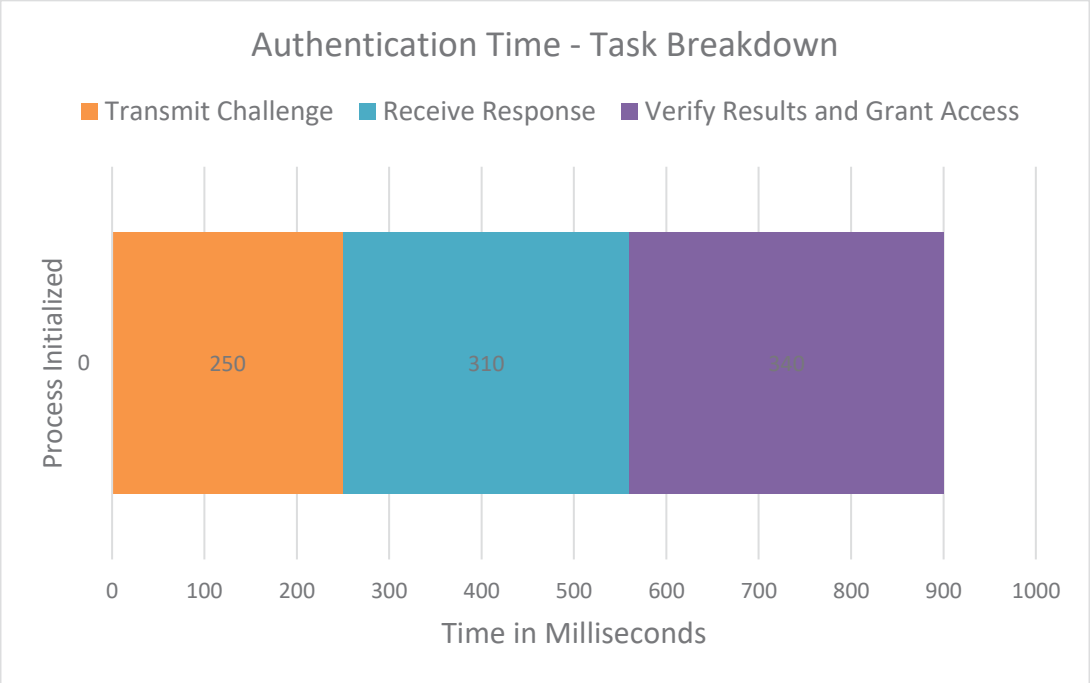

Figure 4.6 Scenario 2 Timeline

Similar to Scenario 1, the simulation for Scenario 2 is also conducted according to the process explained in Section 4.3.2. This simulation is conducted without the presence of the WR and uses the K_{CR} for encryption and decryption. Figure 4.6 displays a timeline of the authentication process starting at '0' seconds and finishing at '0.34' seconds while mapping the key tasks in between.

The performance analysis for Scenario 1 and 2 based on resource consumption is also conducted. Figure 4.7 displays the RAM consumption for both Scenario 1 and 2. Figure 4.8 shows the CPU utilisation.

The CPU and RAM usage depicted provides an overview of the resource utilization for each scenario. As Scenario 1 takes place over a longer period of time in comparison to the faster Scenario 2, the average RAM requirements are comparatively less as the computation is spread across a longer time frame. However, due to the use of asymmetric encryption the average CPU requirements is considerably higher when compared to the requirements for symmetric encryption is Scenario 2.

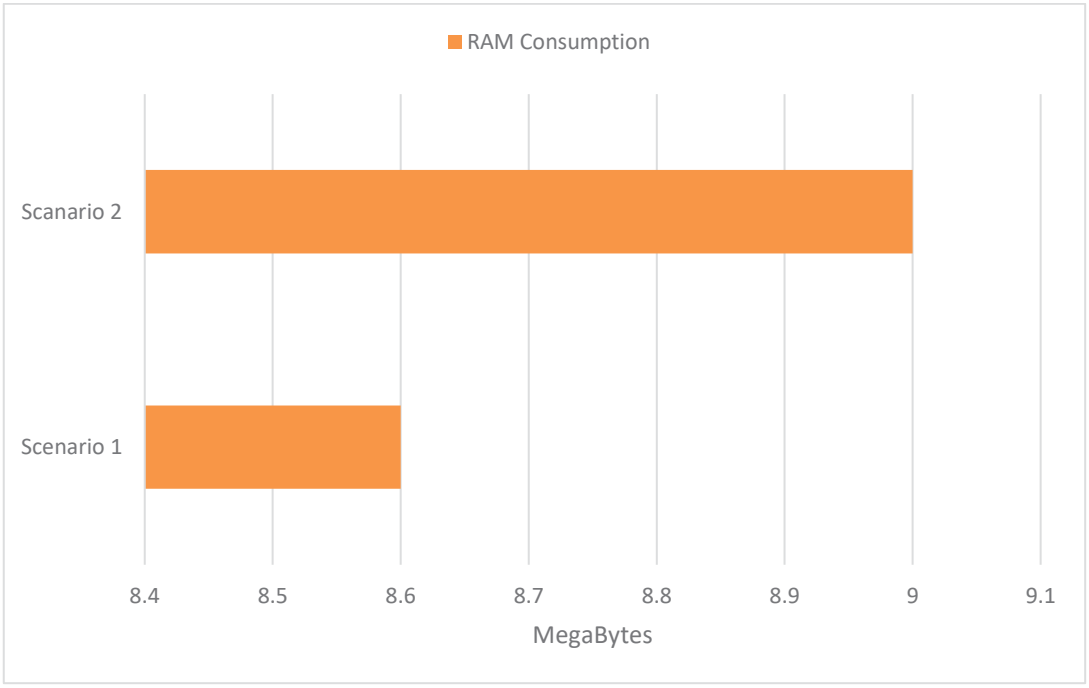

Figure 4.7 Memory Consumption

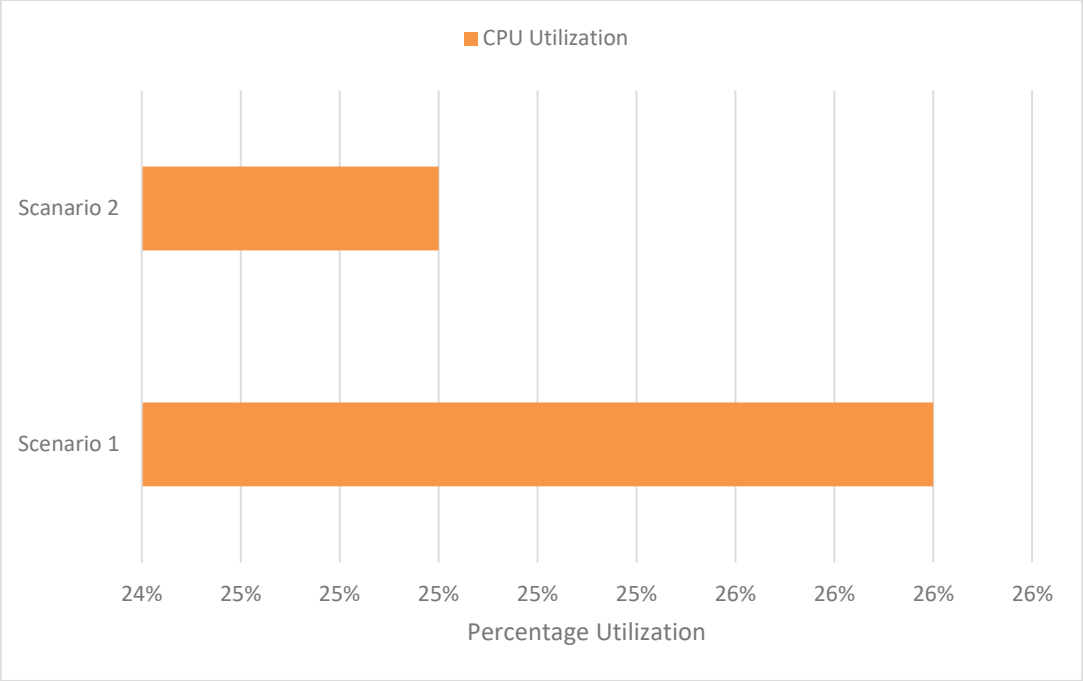

Figure 4.8 CPU Usage

As we can see in the above figures, the full authentication takes almost 3 seconds more than the quick authentication. However, the presence of both scenarios with their conditions together provides better security for the network. In terms of the performance analysis, both the scenarios have similar resource utilisation, which is mainly required for encryption and decryption purposes.

4.5 Comparative Analysis

The hybrid authentication presented in this chapter has been compared with existing approaches based on factors such as the prerequisite, security component and verification process[87]. Table 4.2 provides a summary of the comparison between existing security models and the Secure-GLOR model.

Table 4.2 Comparison Between Security Protocols

Model	Protocol Used	Prerequisite	Security Component	Verification Process	Advantage	Limitation
SRP	DSR	Key shared between the source node and destination node	Messenger authenticati on code	Source address, destination address, messenger ID	Simple algorithm, wide application situations	Lack of protection for routing maintenance messenger, intermediate nodes cannot reply to the routing request
ARIADNE	DSR	Dispatches the TESLA verification key, key shared between the source node and destination node, public key certificates	One-way hash chain messenger authenticati on code	Whole packet, routing sequence	Uses symmetric cryptography and TESLA technology, low computationa l complexity and overhead of management	Needs time synchronization , bandwidth wasted in sending keys, latency in verification
ARAN	AODV	Establishes a certificate server responsible for issuing and maintaining the public key certificate of every node	Digital signature	Whole packet	Ensures authenticatio n, integrity, and nonrepudiatio n	High computational complexity, CA is needed, intermediate nodes cannot reply to the routing request
SEAD	DSDV	Dispatches the verification initialization value	One-way hash chain	Sequence number, number of hops	Low complexity in computation	A trusted entity is needed to dispatch and maintain the verification elements of all nodes
SAODV	AODV	Dispatches public keys of nodes	Digital signature, one-way hash chain	Whole packet	Intermediate nodes could reply to the routing request	High computational complexity due to the asymmetric cryptograph

SLSP	ZRP	Dispatches public keys of nodes	Digital signature, one-way hash chain	Whole packet	Prevents DoS attacks by monitoring neighbour nodes	High computational complexity due to the asymmetric cryptograph
Secure-GLOR	GLOR	Registers node and public key in web-register	Challenge-Response	Source ID, source public key	Adapts to the situation, can provide authentication without web-register	A device must be registered, have a valid ID and a linked public key.

4.6 Summary

Dynamic wireless mesh network is an emerging technology in the area of self-sustained networks and holds the key to evolve into the next generation communication network. However, it is limited only by the static protocols and rigid security frameworks preventing it from evolving and are hence not suitable for the dynamic network.

The dynamic wireless mesh network requires a new routing protocol such as the GLOR and security models that are flexible and can adapt to various scenarios faced by the dynamic network. The hybrid authentication scheme presented in this chapter is one such aspect, which works according to the network rather than have the network work according to it.

Along with the flexibility, the security model also uses new methods to provide higher levels of security as mesh networks are prone to various attacks as discussed in Section 2.5. With additional security features (as elaborated upon in the chapters ahead) being added to the Secure-GLOR model, the dynamic wireless mesh network can become better managed, more secured and scalable.

Chapter 5

Encryption Techniques for Dynamic Distributed Networks

As we progress into a digital era where most aspects of our lives depend upon a network of computers, it is essential to focus on digital security. Each component of a network, be it a physical network, virtual network or social network requires security when transmitting data. Hence the dynamic wireless mesh network must also deploy high levels of security as found in current legacy networks. This chapter presents a secure Geo-Location Oriented Routing (Secure-GLOR) protocol for wireless mesh networks, which incorporates a hybrid encryption scheme for its multilevel security framework. The hybrid encryption technique improves the network's overall performance compared to the basic encryption by using a combination of symmetric key as well as asymmetric key encryption. Using the combination of the two encryption schemes, the performance of the network can be improved by reducing the transmitted data size, reducing computational overhead and faster encryption-decryption cycles. This chapter discusses multiple combinations of encryption schemes for both symmetric and asymmetric encryption

and compares their performance in various experimental scenarios. The proposed security scheme achieves better performance based on the results obtained with most viable options for our network model.

5.1 Introduction

With the advancements in technology, the digital world has a greater impact on various aspects of our lives and has become an essential part of our daily life. The communication network is one of the most important parts, keeping us connected to the digital world. We rely on the communication network to access the internet, control traffic, make payments and even carry our sensitive data, hence it is essential that it maintains high levels of security to ensure the integrity of the network.

The continuously rising need for security is now expanding to every type of network, be it social or physical. This need for security has also come to wireless mesh networks that have been in development over recent years. The mesh networks are known for their ability to form self-sustained and easily configurable networks by connecting a large number of devices together; however, guaranteeing security in such networks is one of the major challenges for future application-specific deployments. Unlike the legacy networks, the mesh networks depend on their devices to relay the data by sending it through a chain of devices, which means that the data is accessed by more than just the device it was destined for. Hence the need for secure delivery of data is very critical to the future of such a network model [85].

In addition, the dynamic wireless mesh networks require a custom-tailored security framework as the current/legacy security solutions do not fit well and limit their capabilities. This is because the present security solutions for a mesh network requires a central entity dependent network which limits the size and dynamic-ness

when applied to the wireless mesh network. In this chapter, we present the Secure-GLOR model based upon the GLOR protocol. The Secure-GLOR model is specifically designed to embrace the dynamic properties of a wireless mesh network while providing a high level of security.

This chapter introduces the hybrid encryption scheme for the secure version of the GLOR protocol, as proposed in our papers [19, 22]. Section 5.2 of this chapter begins with a discussion about existing approaches/models and how they implement security. The security model and its various aspects implemented by the GLOR protocol is then explained in Section 5.3 followed by a theoretical analysis of the model in Section 5.4. Section 5.5 presents the performance of the network model under different scenarios with various configurations and discusses the results obtained. Finally, Section 5.6 summarizes with the final thoughts on the next step of the Secure-GLOR protocol.

5.2 Existing Models

Amongst the models/approaches explained in Section 2.3, very few take into account the security of data being transmitted. The Smart Phone Ad-hoc Networks (SPAN) project [64] was the earliest practical implementation showing an off-grid network; however, the project had no current security implementation. Though it discusses the use of public-private key pair for encrypted communication between devices, the key exchange process was manual and a major risk.

Several Project [68] and FireChat [73] are other similar implementations which use Wi-Fi/Bluetooth to create a self-sustained network. However, the methodology lacks security as each message is sent to every device on the network without any encryption, like a chat room. Other implementations such as the BRIAR Project [74] have been designed to provide secure and resilient peer to peer communications

with no centralized servers and minimal reliance on external infrastructure. This is however achieved with a loss of real-time transmissions.

There are several security threats in wireless communication networks, the layer-wise classification of the security threats and solutions are given in [88, 137]. In [138, 139], it is already proved that symmetric key solutions are very many times faster than asymmetric key solutions. Symmetric key cryptography is always suitable for the low power devices, where a shared key needs to be updated after a certain period of time [138-140].

Current research trend creates hybrid architecture by combining communication and computing technologies such as fog or cloud computing. In [90, 141, 142], authors have given the novel security solutions for these hybrid architectures. By following the hybrid architecture, we have applied both symmetric key cryptography [102, 103] and asymmetric key cryptography [108, 128] for our proposed GLOR protocol.

5.3 Encryption Techniques

An essential part of a network's security depends upon how best it can secure the data while it's being sent across the network. The Secure-GLOR model was initially designed to use a standard asymmetric encryption algorithm to avoid refreshing encryption keys frequently. However, after extensive testing and multiple simulation performance analysis, the model has since been updated to use a hybrid encryption technique to overcome the delays caused when transferring bigger chunks of data. In this section, we begin with discussing the initial standard design that incorporated asymmetric encryption for all tasks and how it performed as compared to the hybrid encryption model. Table 5.1, as shown below, lists the various notations used.

Table 5.1 Notations Used

Notation	Component	Description
Node	Network Device	A device with an established connection to the network and is authorized to authenticate other devices.
Device	New Device	A device which wishes to join the network.
WR	Web Register	A database that stores network device information such as Unique ID, MAC, Address, Public Key, etc.
UID	Unique ID	A unique identifier generated and provided to each node by the Web Register. It is linked with each device's MAC.
ADDR	Geo-Location Address	Physical position (two dimensional) of the device determined through its latitude and longitude coordinates
K_{PU}	Public Key	Asymmetric encryption key pair used for authentication and End-to-End encryption. Each device gets its own key pair.
K_{PI}	Private Key	
K_{SE}	Session Key	Symmetric encryption key provided to each device at registration.

5.3.1 Asymmetric (Standard) Encryption Technique

In the asymmetric encryption technique [22] Secure-GLOR used RSA to encrypt the message part of the smart packet. It begins during the packet formation process after the origin node requests the details of the destination node as described in detail in Chapter 4. To summarize, each node generates a new public-private key pair during its first registration. Once the registration process is completed successfully, the node sends a copy of the public key to the web register. This is used when a node wishes to send some data to another node on the network.

Algorithm 5.1 Key Management

$K_{PI}(i)$ – Private Key of node *i*
$K_{PU}(i)$ – Public Key of node *i*
WR - Web Register; SN - Source Node; DN - Destination Node

1. At initial node registration
   ∀ node *i* generates its key pair i.e. $K_{PI}(i)$ and $K_{PU}(i)$

2. Nodes share own public key with web register (WR)
   $K_{PU}(i)$ → WR

3. During data transmission
   SN (request) → WR (DN location)
   If WR authenticate SN and found DN in the register
   WR → SN: ($K_{PU}(DN)$ ‖ *Loc (DN)*)
   Then SN uses $K_{PU}(DN)$ for data encryption.

As explained in Section 3.3.4, once the node receives information about the destination node, it also receives the public key of the destination node. This is used to encrypt the message part of the packet such that only the destination node can decrypt it using its own private key. In addition to encrypting the message, the node also provides its own public key that the destination node can use to encrypt any response it wishes to send. The complete procedure for key management in Secure-GLOR is shown in Algorithm 5.1.

5.3.2 Hybrid Encryption Technique

During the evaluation of different metrics in the simulation performances (Section 5.5.3), it was observed that the asymmetric encryption technique performed similar to the symmetric encryption technique with a small dataset. However, during the implementation and testing phase, it was discovered that symmetric encryption algorithms performed considerably better compared to the asymmetric encryption algorithms when a larger sized data set was used. Due to the

fact that asymmetric algorithms use a larger key size to provide the same amount of security as compared to symmetric algorithm keys and also due to the workings involved in the encryption-decryption process for both, the asymmetric encryption techniques are more resource and time intensive as compared to symmetric encryption techniques that are able to provide the same/better level of security.

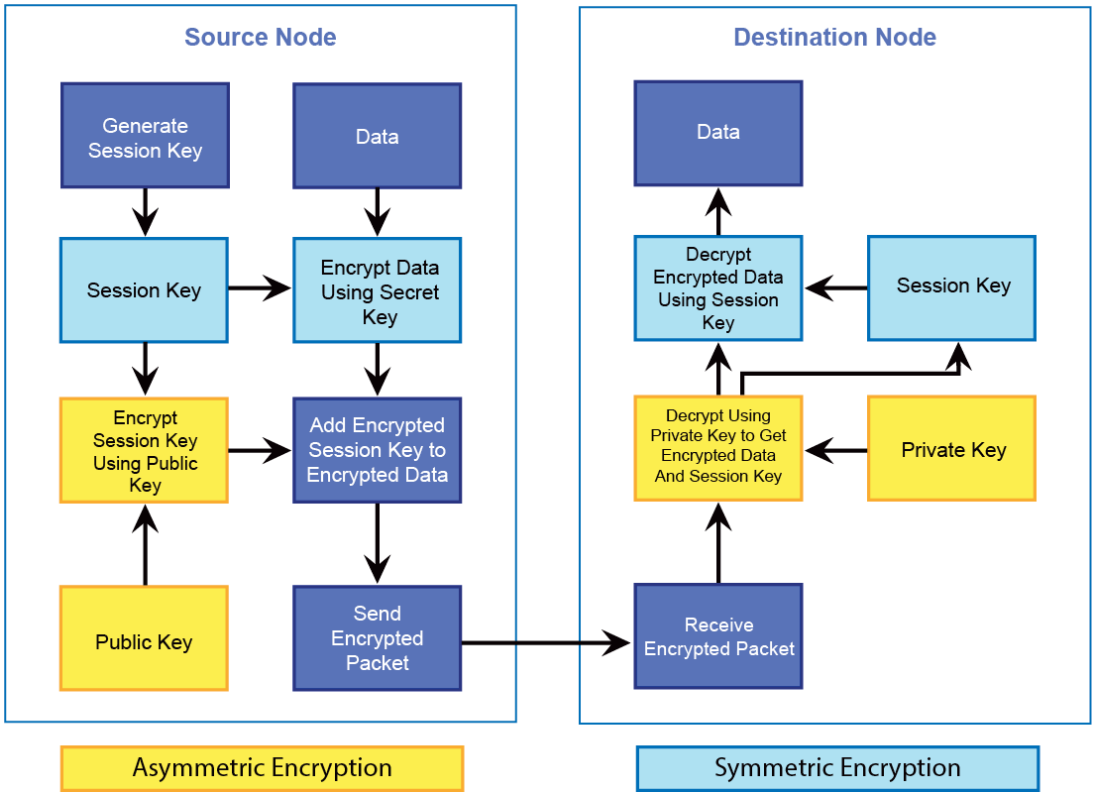

Figure 5.1 Hybrid Encryption

In order to provide better and faster encryption/decryption cycles for the Secure-GLOR model, we use the hybrid encryption scheme. According to the hybrid encryption scheme, a symmetric encryption technique is used to encrypt the data while the asymmetric encryption algorithm is used to encrypt the key used by the symmetric encryption. The essence of hybrid encryption is the use of a session key (symmetric encryption) which is randomly generated and exchanged between the source node and destination node securely.

To ensure the session key is not stolen or tampered with, the asymmetric encryption is used to encrypt the session key using the public key of the destination node ensuring that only the destination node can access the session key and use it to decrypt the data as shown in Figure 5.1. The process for session key generation and key management is also shown in Algorithm 5.2.

Algorithm 5.2 Session Key Management

$K_{PI}(i)$ – private key of node *i* $K_{PU}(i)$ – public key of node *i* K_{SE}(i) – session key WR - web register; SN - sender node; DN - destination node

1. At initial node registration
   ∀ node *i* generates its key pair i.e. $K_{PI}(i)$ and $K_{PU}(i)$
2. Nodes share own public key with web register (WR)
   $K_{PU}(i)$ → WR
3. Before Data transmission
   SN (request) → WR (DN location)
   If WR authenticate SN and found DN in the register
   WR → SN: ($K_{PU}(DN)$ ‖ *Loc (DN)*)
4. Generate Session Key
   SN(Random) → K_{SE}
   SN uses K_{SE} for data encryption.
5. Session Key Encryption
   SN uses $K_{PU}(K_{SE})$ for session key encryption.
   Encrypted (data & K_{SE}) are concatenated and sent

5.4 Theoretical Analysis

This section provides a theoretical analysis of our proposed Secure-GLOR model to demonstrate its response to potential security threats and how Secure-GLOR is protected against them. We use hybrid encryption where asymmetric key

cryptography is used to protect the session key and symmetric key cryptography is used to protect data in the dynamic mesh network. The proposed security method performs efficiently without degrading network performance. We have made the following practical and realistic assumption in our method:

Assumption 1: In our method, the data that is encrypted by a symmetric-key method cannot be decrypted by any other, unless they have the session key.

Assumption 2: In our method, the session key that is encrypted by an asymmetric-key method cannot be decrypted by any other, unless they have the private key.

5.4.1 Security Proofs

Definition 1 (attack on integrity): A malicious attacker M_I can attack the integrity if it is an adversary capable of monitoring the data packets regularly and trying to access and modify them before they reach their destination.

Definition 2 (attack on confidentiality): A malicious attacker M_C is an unauthorized party which has the ability to access or view the unauthorized data packets before they reach the destination node.

Theorem 1: Proposed Secure-GLOR maintains end-to-end security in a mesh network with dynamic nodes.

Proof: We use an asymmetric key cryptographic method to maintain end-to-end security over our Secure-GLOR protocol in a dynamic wireless mesh network. As our network model is comprised of high-performance devices (i.e. smartphones, laptop, etc.), we aim to reduce the number of keys that are in use, and in doing so, reduce the overall network overhead.

In symmetric key cryptography, with N number of nodes, the number of pairwise keys required for a secure communication can be calculated as shown below:

If a new node 'I' is added to the network, it must generate and share a new key with its neighbouring nodes.

Then for N users, the number of keys required is $1 + 2 + \cdots + (N-1) = \frac{N(N-1)}{2}$

⇒ there will be $O(N^2)$ keys.

Using a similar method, the number of pairwise keys required for asymmetric key cryptography with N number of nodes can be calculated as below:

If a new node 'I' is added to the network, it needs only a public key and a private key to share a new key with the WR as well as other nodes.

Then for N users, we have $2N$ keys.

⇒ there will be $O(N)$ keys.

While comparing with other existing symmetric key algorithms, individual nodes may require a separate pair, so as a result, we have 4N keys i.e. O(N) keys.

Another advantage with the asymmetric key over the symmetric key algorithm is that it does not require changing or updating the key after certain intervals of time, which leads to reduced network communication overhead and loss of secret keys. Our security method uses a public key (K_{PU}) to encrypt and private key (K_{PI}) to decrypt the session key (K_{SE}), and each node only shares its public key with the web register. Hence, an intruder might reach the web register to obtain the public key but it is impossible to get the private key without compromising the node as its private key never leaves the device.

Finally, only the recipient node can decrypt the packet using its own private key (K_{PI}) to get the session key and use it to decrypt the data. Therefore, we can conclude that Secure-GLOR maintains end-to-end security.

Theorem 2: Secure-GLOR is secure against an attack on integrity and confidentiality

Proof: Following Algorithm 5.1, it is clear that the intruder cannot get the destination node's private key to decrypt the data packet.

From Definition 1, we know that an attacker M_I has full access to the network to read data flow, however, M_I cannot get the private key information of the destination node. The intruder can gain access to the public key K_{PU} but it's useless as there exists no such method to obtain/derive the private key using the public key. In the same way, following Definition 2, M_C can gain access to the public key K_{PU} but no other information.

Finally, we can conclude that both M_I and M_C can neither read nor modify the data packets, the data packet can only be accessed by its destination node. Hence, Secure-GLOR is secure against an attack on integrity and confidentiality.

5.4.2 Forward Secrecy

By following a standard asymmetric key cryptography procedure, destination node's public key is used to encrypt the session key which in turn encrypts the data packets, hence it can only be decrypted using destination node's private key to first decrypt the session key and using it decrypt the data. Even if the public key is known to intruders, it cannot be used to decrypt the packet. We choose to use asymmetric key cryptography over symmetric key cryptography because network nodes have enough resources, battery and computational power to compute complex encryption/decryption. This introduces technical challenges for the intruder to break

the encrypted data packets. This also avoids a repeated rekeying process and reduces communication overhead.

The proposed Secure-GLOR method is secured against any kind of malicious attack as we use different keys for encryption and decryption process. Finally, we conclude that an intruder cannot predict the keys to read the data packets.

5.5 Results and Validation

As discussed in Chapter 3 and paper [18] the GLOR protocol has been developed in Visual Studio using C#. The machine used for simulation is powered by a 6th Gen. Intel i7 (3.1 GHz) CPU and 16GB DDR3L RAM running Windows 10.

5.5.1 Simulation Environment Setup

The environment consists of nodes evenly spread on a 2D plane. The nodes location is calculated using the X-Y coordinate of the device on the 2D plane. The web register is implemented using a local database. The nodes have been allocated random transmission speeds varying from 11Mbps to 25Mbps, based on which the transmission time is calculated.

As the aim is to test absolute performance of different encryption algorithms using the same conditions, the test-bed includes the following assumptions

- The nodes have already been authenticated and have a unique id.
- None of the nodes fail during the operation.
- All nodes have the capability to calculate their location.
- No packet is dropped during the transmission process.
- Each node has a direct/indirect connection to the web register.

5.5.2 Simulation and Observation

The simulation initiates with the nodes calculating their geo location (using their X-Y coordinates) and generating a Public-Private Key pair for session key encryption and a random session key for data encryption. The nodes send their location and public key to the web register and start connecting to the neighbour nodes to create the neighbour table to improve network performance.

The nodes use a predefined data-set to be communicated between the source and destination nodes. There are 72 nodes being used in this setup and information like transmission time, CPU utilization, and memory utilization is calculated and compared with various encryption schemes. This provides us with valuable information about how the network performs under different scenarios.

5.5.3 Results and Analysis (Standard Encryption Technique)

Figure 5.2 and Figure 5.3, give us a network performance insight with respect to the time taken for a data packet to be created, sent to the destination and receive an acknowledgement for the same. It also shows us the amount of delay obtained in proportion to the distance travelled.

The comparison consists of both symmetric encryption (AES 128, AES 256) and asymmetric encryption (RSA 2048, RSA 4096). In the first scenario, a data-set of 500 Bytes is encrypted and sent from the origin node to the destination node. From Figure 5.2, it is observed that there is a steady increase in the time taken for the packet to reach its destination and it is directly proportional to the distance travelled (number of hops). The graph also shows that there is very little difference in the time taken by AES 128 and RSA 2014; however, AES 256 and RSA 4096 take a comparatively longer time due to the increased size of encrypted data. This implies

that the network is able to perform normally even after encryption is used and does not cause any overhead/overload on the devices.

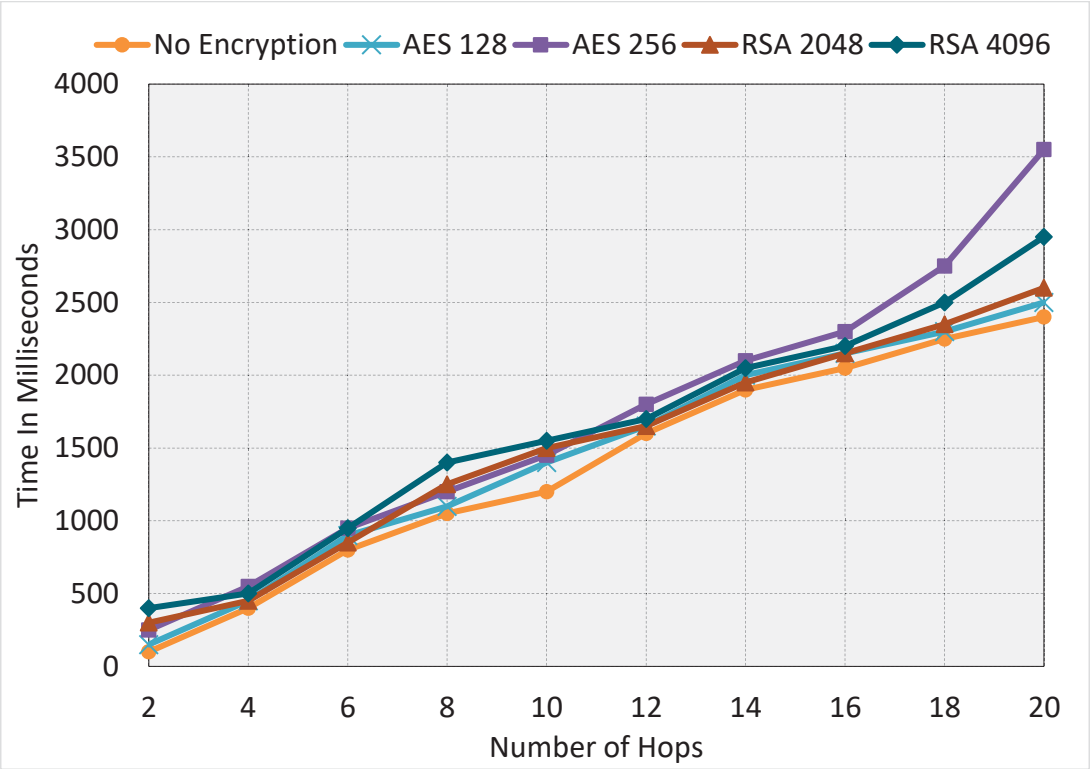

Figure 5.2 Time Taken for the Trip (500-Bytes Data)

In the second scenario, a data-set of 64000 Bytes is used to test the simulation. The results, as shown in Figure 5.3, depict that the symmetric encryption has a similar steady increase comparable to scenario one. However, the asymmetric encryption has a very large increase (average 13 seconds for RSA 2048 and 42 seconds for RSA 4069).

The major factor for such a high increase can be directly related to the key size for RSA encryption, which requires the data to be broken down into small chunks and then encrypted individually. This leads to a longer wait cycle during encryption and decryption process (at the origin and destination node).

Figure 5.3 Time Taken for the Trip (64000-Bytes Data)

The performance analysis of each encryption technique based on resource consumption was also carried out in the above-mentioned scenario one and two. As Figure 5.4 and Figure 5.5 show us, the symmetric encryption techniques had similar memory consumption of about 4 Megabytes, however, AES 256 had almost double CPU usage of 24 % as compared to AES 128 which only required 14%. Hence it can be deduced that even though the time required for both techniques is almost identical, the resource requirement for AES 256 is much more.

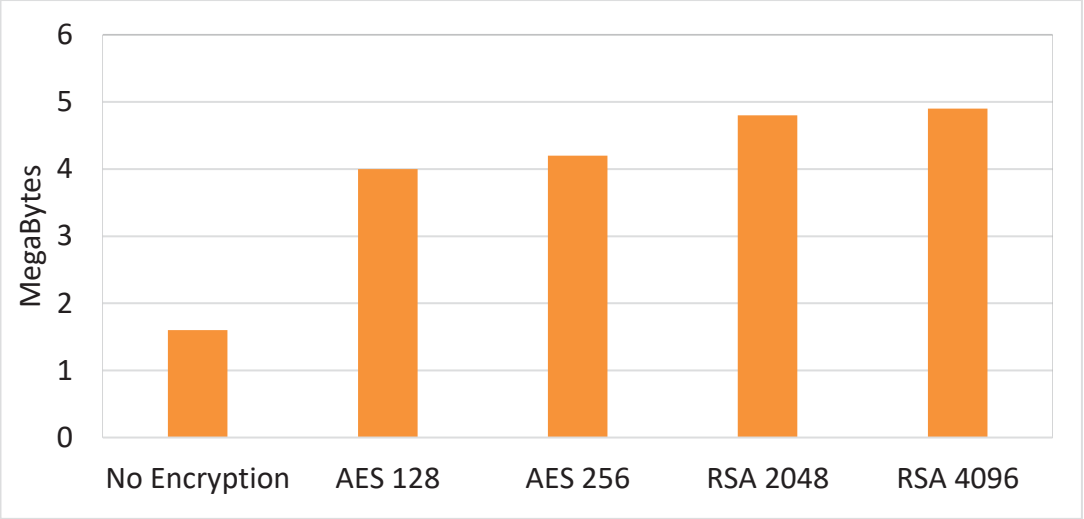

Figure 5.4 Memory Consumption

The asymmetric encryption techniques RSA 2048 and RSA 4096 had similar memory consumption (4.8 Megabytes) and CPU usage (27%-28%). Hence both techniques require almost the same amount of resources, but their time consumption is directly proportional to the amount of data. However, overall, the symmetric encryption had less resource consumption when compared to asymmetric encryption.

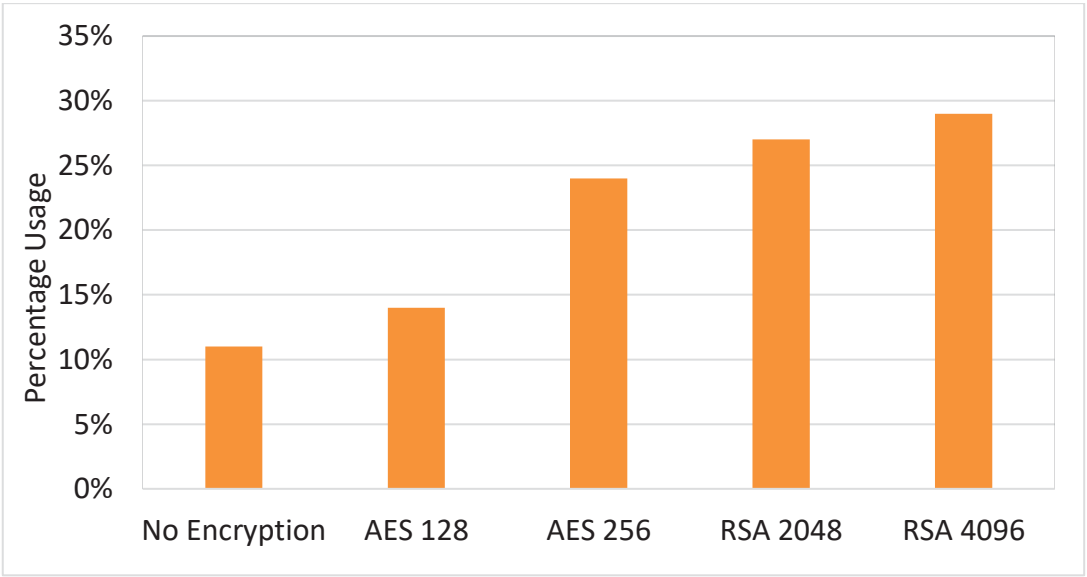

Figure 5.5 CPU Usage

5.5.4 Results and Analysis (Hybrid Encryption Technique)

In the previous section, results were calculated for both symmetric and asymmetric encryption techniques individually. The same constants are now used to implement the hybrid encryption. This section presents and compares the results from the hybrid encryption techniques.

As discussed above, the hybrid encryption technique uses a combination of symmetric and asymmetric encryption schemes. The symmetric encryption technique is used to encrypt the data using a session key, whereas the asymmetric encryption technique is used to encrypt the session key using the public key of the destination node.

We have considered the following encryption techniques based on their individual strength and performance. We evaluated AES 128, AES 256 and Blowfish for the data encryption while in combination with RSA 1024, RSA 2048 and Elliptic Curve Diffie Hellman (ECDH) for session key encryption. The combination of symmetric and asymmetric techniques used have been referred to as “case” and are listed in Table 5.2.

Table 5.2 Hybrid Encryption Scenarios

Case	Symmetric Encryption	Asymmetric Encryption
0	None	None
1	AES 128	RSA 1024
2	AES 128	RSA 2048
3	AES 128	ECDH
4	AES 256	RSA 1024
5	AES 256	RSA 2048
6	AES 256	ECDH
7	Blowfish	RSA 1024
8	Blowfish	RSA 2048
9	Blowfish	ECDH

Figure 5.6 and Figure 5.7 give us a network performance insight with respect to the time taken for a data packet to be created, sent to the destination and an acknowledgement received for the same. It also shows us the amount of delay obtained in proportion to the distance travelled. As the constraints are kept exactly as before, we take a 500-byte data-set for the first setup (Scenario 1) and a 64000-byte data-set for the second setup (Scenario 2).

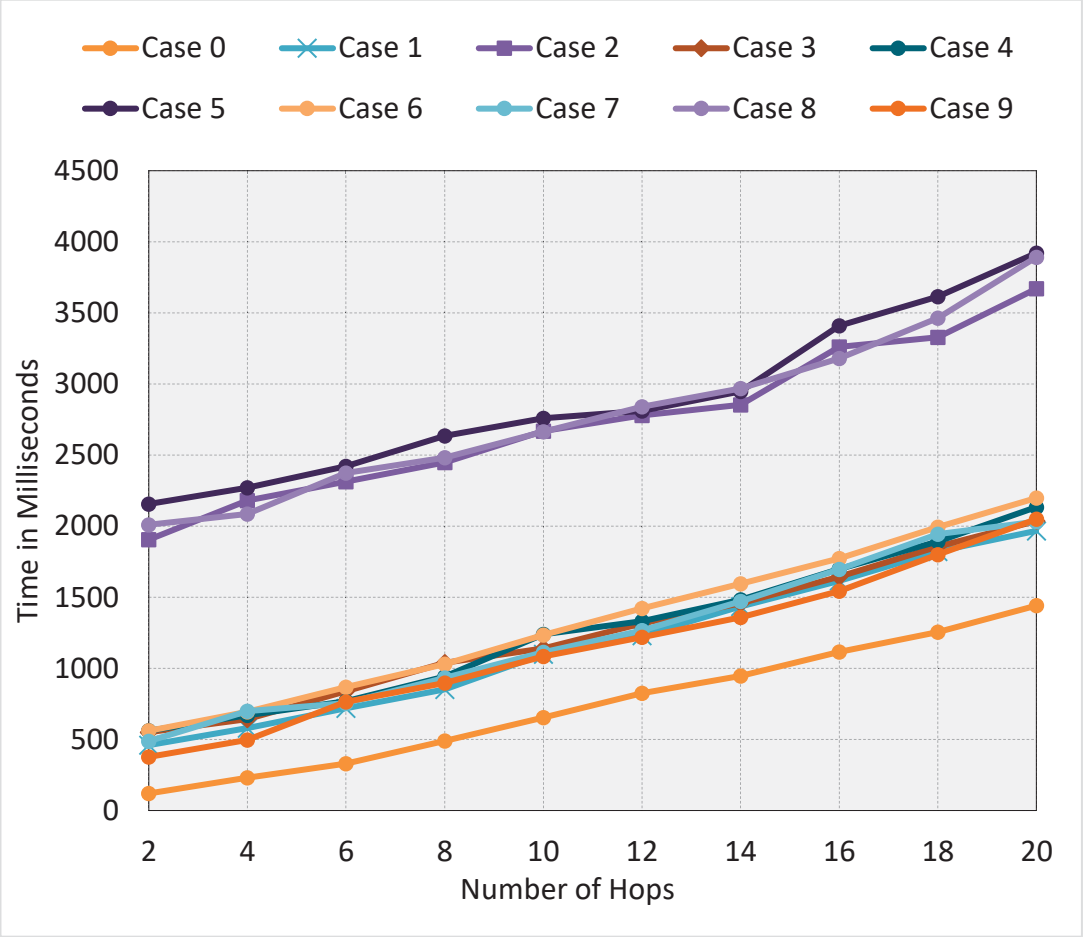

Figure 5.6 Time Taken for the Trip (500-Bytes Data)

For Scenario 1, using a 500 bytes data-set, it is observable from the data presented in Figure 5.6 that hybrid encryption techniques using RSA 2048 (Case 2, 5 & 8) take up to 2000 milliseconds extra as compared to the RSA 1024 or ECDH for secret key encryption. Another closer observation also tells us that the hybrid

encryption techniques using AES 256 (Case 4, 5 & 6) take slightly more time to encrypt the data as compared to AES 128 of Blowfish. In general, a gradual increase in time is expected with a direct relation to increase in number of hops a packet must travel before completing a round-trip.

When re-running the simulation for Scenario 2 with a data set of 64000 Bytes data-set, the results obtained show an increase of approximately 250 milliseconds for all cases as compared to Scenario 1. This upward shift is directly proportional to the size of data being encrypted and sent over the network. A gradual increase in time is once again expected with a direct relation to increase in number of hops similar to Scenario 1.

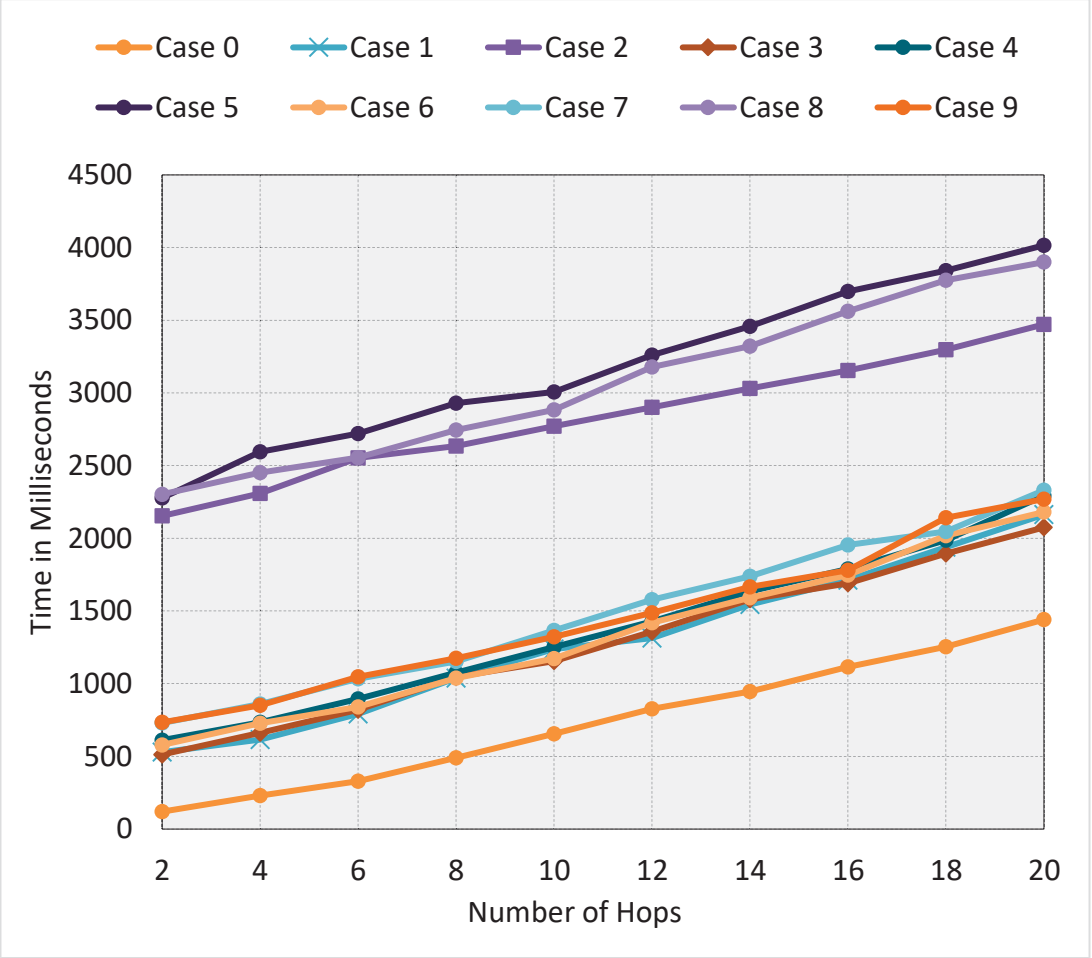

Figure 5.7 Time Taken for the Trip (64000-Bytes Data)

Other observations such as the increased 2000 milliseconds it takes for RSA 2048 as compared to RSA 1024 or ECDH and approximate incremental of 250 milliseconds for AES 256 as compared to AES 128 of Blowfish is also similar in both Scenario 1 and Scenario 2. Overall the data from both Scenarios can be majorly classified on the basis of hybrid encryption using RSA 2048 and hybrid encryption using RSA 1024 / ECDH due to the huge difference.

As just the time parameter is not enough to select a preferred combination of symmetric and asymmetric encryption technique, we also take into account the performance of the encryption techniques by measuring the resources used. The performance analysis is another major factor in deciding how much resource consumption is adequate for the network model. Figure 5.8 displays the Memory (RAM) consumption and Figure 5.9 displays the CPU utilization for the various cases discussed above.

Figure 5.8 Memory Consumption

From Figure 5.8 we can observe that the hybrid encryption using AES 256 has the highest RAM consumption for data encryption as compared to other symmetric

encryption techniques. It can also be deduced that RAM consumption is also high when using RSA 2048 for session key encryption as compared to other asymmetric encryption techniques. Figure 5.9 also presents similar results for maximum CPU consumption by the RSA 2048 when encrypting the session keys. However, the highest CPU consumption for encryption of the data is AES 128, which is unlike the previous observation.

Figure 5.9 CPU Usage

In conclusion, when compiling all the data received it is clear that Case 9 has the least resource consumption as well as one of the best performance amongst the compared symmetric and asymmetric encryptions techniques. Hence it seems promising to use Blowfish as the symmetric encryption technique for the encryption of data using the session key and for encrypting the session key Elliptic Curve Diffie Hellman (ECDH) would be the preferred asymmetric encryption technique.

5.6 Summary

The results collected from the various combinations of symmetric encryption techniques and asymmetric encryption techniques were vital in deciding the preferred combination to use for the hybrid encryption model. The choice to use hybrid encryption also improves the performance of the network by reducing the load on the devices caused if only asymmetric encryption was used.

This is because of the fact that asymmetric encryption can only encrypt a certain amount of data at a time which is dependent on the size of the keys being used and the resource intensive processes involved. Hybrid encryption overcomes this issue by using symmetric encryption, which has been proven to be a faster technique, to encrypt the data using a session key which in turn is encrypted by an asymmetric encryption technique.

In addition to the hybrid encryption technique, the Secure-GLOR also benefits from the hybrid authentication (discussed in Chapter 4) scheme applied across network devices. This enables us to create a complete integrated security model for a mesh network. In the next chapters, we will discuss further about the dynamic key management strategy to defend against multiple attacks and threats.

Chapter 6

Key Management Scheme for Distributed Networks

As distributed networks, such as the dynamic wireless mesh network, evolve, they expand in both the coverage they provide and the number of devices they can support. However, with such an expansion, providing security for such networks has become a major concern. The most important component of a security framework is establishing the keys to be used in the network. This has always been a challenge for distributed networks as the network expansion makes it difficult to keep setting up authorities to aid in the key-exchange process. In this chapter, we address the challenge of key-exchange in distributed networks by presenting a new scheme which provides a safe method of key exchange using a mix of random selection, multi-path approach and anonymity. The proposed approach is thoroughly analysed to measure its strength and effectiveness against various threat scenarios.

6.1 Introduction

As distributed networks are decentralized, they require a robust scheme to preserve the security of the devices connected to them. In the last decade, an increasing number of devices have become a part of such networks and the trend is still growing. One of the main reasons for the increased popularity of distributed networks like the wireless mesh network is the fact that they can be set up anywhere with minimal infrastructure requirement. However, with an increase in unattended standalone networks in adversarial environments, the need for improved security is very high [143, 144]. In this chapter, we present a new key-distribution scheme specifically designed for distributed networks.

Although the security of such networks is easily improved using end-to-end encryption, the most crucial component of the process is the key exchange as it provides the key to be used for the encryption/decryption process. The key-exchange can be attempted using various schemes, each one with its own set of keying style [118] as depicted in Table 6.1. The approach to be used for a specific network is determined after analysing the various components and configuration of that distributed network.

The encryption/decryption process can be conducted using a symmetric key where a master key is distributed throughout the network. However, if even one of the device on the network is compromised, it will compromise the whole network [145]. To avoid this, individual asymmetric or public-private key pairs are preferred as the public key can only be used to encrypt and only its linked private key can decrypt. This way even if a device is compromised, it would not be able to compromise the whole network.

These keys (or keying material) can be pre-loaded onto the devices (key pre-distribution) and used when connecting to the network. This approach is however

very static and can disrupt the network by compromising enough devices to be able to compromise other keys being distributed [117]. To avoid such scenarios, dynamic key-pairs are used and can be updated regularly to maintain a higher level of security [146, 147]. In addition to the approach and keying style, there are certain constraints which need to be considered when designing a key distribution scheme for distributed networks [120].

Table 6.1 Types of Key Distribution Approaches

Approach	Mechanism	Keying Style
Probabilistic	Pre - Distribution	Random-Key
		Pair-Wise key
Deterministic	Pre – Distribution	Pair-Wise key
		Combinatorial
	Dynamic Key Generation	Master Key
		Key Matrix
		Polynomial
Hybrid	Pre – Distribution	Combinatorial
	Dynamic Key Generation	Key Matrix
		Polynomial

The devices that form a distributed network can span over very large areas and hence communicate using the concept of hopping; this implies that any data sent over the network is travelling through other devices before finally reaching its destination. The distributed networks also contain mobile nodes that keep changing locations, and in doing so, also change the network layout [121]. A major factor on which the key strength and size depend is the computational resources available with the devices in the networks [119].

To address the above concerns and constraints, we present our proposed Multi-Path Anonymous Randomized Key (MPARK) distribution scheme. Section 6.2 discusses the assumptions made regarding the applicable network structure and provides the notations used. The chapter then presents the MPARK distribution scheme and its components in Section 6.3. In Section 6.4 we expand upon the various

security threats and how they are overcome using our scheme. Finally, we conclude our proposed scheme and discuss its future development in Section 6.5.

6.2 Assumptions and Notations

This section states some realistic assumptions for the distributed network scenario where our proposed key distribution scheme MPARK can be applied. This section also defines the notations used while defining the proposed scheme.

6.2.1 Network Assumptions

The MPARK distribution scheme considers some realistic assumptions to maintain an adequate level of security. As it is an integral part of a network, there must exist a central support entity such as the web register defined in Section 3.3.3.1. The web register is able to keep a record of all the public keys in use and the devices they are linked to. The web register also has enough processing power to conduct complex calculations and fast storage to traverse and look up data quickly. The web register must always be available having a distributed presence with appropriate redundancies to prevent single-point failures. It must also be able to maintain an active connection to the network devices directly or through a multi-hop approach.

Another integral part of a distributed network is the devices it is formed of; hence it is important that all the devices have adequate resources. The devices must have enough processing power to be able to generate their own asymmetric key pairs (used in key exchange) and symmetric key (used as a session key). In addition to adequate processing power, they must also contain enough memory to store a few megabytes of data such as the key pool and keying material.

A new device that is connecting or seeking connection to the network must be authenticated before the key exchange is initiated. Until the authentication process is complete, the new device must not be given full or partial access to the network. In order to maintain the integrity of the proposed scheme, it is also important that the new device is within range of two or more edge devices (a device which is already a part of the network) before the key exchange is initiated.

6.2.2 Notations Used

Table 6.2. lists the various notations used, the component they refer to and its description. These notations are used to refer to integral components of the proposed scheme.

Table 6.2 Notations Used

Notations	Component	Description
$\mathbf{D_N}$	New Device	A new user device that is attempting a key exchange.
$\mathbf{D_E}$	Edge Device	A user device which is currently part of the network and has an existent key pair set up with the web register
Server	Web Register	A centralized system used to keep a record of public keys and the devices they correspond to. A user device can use it to look up the public key for another device it wishes to contact.
$\mathbf{K_{PU}}$	Public Key	A unique key linked to a user device's private key. It is publicly stored on the web register and can be used by any device only to encrypt data destined for its user device
$\mathbf{K_{PI}}$	Private Key	A unique secret key stored securely on the user device. Only a specific private key can be used to decrypt data that was encrypted with its linked public key.

K_P	Key Pair	A set of Private Key and Public Key that are linked to each other.
Pkt	Data Packet	A normal TCP/IP packet used to encapsulate data and other important information while it travels from the source device to the destination device.
TTL	Time To Live	A time value which represents the validity of the data packet. If the time to live is expired, the data packet will be discarded.
CON	Active Connections	An active connection refers to an ongoing connection between a device and its in-range neighbour device.
K_{POOL}	Pool of Keys	A collection of Public Keys / Private Keys
IV	Initialization Vector	A random starting variable used as an initializer for the encryption process.
N_P	Number of K_P	The total number of key pairs created during a single key distribution process
N_E	Number of D_E	The number of edge devices involved in the Key Distribution process.
N_S	A set of K_P	The set of Key Pairs (Can also refer to Public Keys in general) that are part of a single set of Keys packed in a data packet
Rnd	Randomizer	An absolute (or pseudo) randomness generator
S_V	Seed Value	The initial elements used as a reference when creating key pairs.

6.3 Proposed Key Distribution Scheme

This section presents the proposed Multi-Path Anonymous Randomized Key (MPARK) distribution scheme to address the issues stated in Section 6.1. The aim is to use anonymity and randomness to further reduce the risks involved in the key exchange over an insecure wireless channel in a mesh network. This channel can be defined as the connection between two devices before a public key or a session key

is used to encrypt the traffic. This section discusses the working of the MPARK distribution scheme, its components, the methodology used and its effectiveness.

The proposed scheme has been divided into its major components to provide its key details. Each component provides its unique features and when combined all together, form a strong key distribution scheme. The major components are individually discussed below.

6.3.1 Initial Contact

The initial contact can be defined as the moment a new device is being added or has requested to be added to the network. During this phase, the new device undergoes the authentication process. The steps involved in the authentication process will vary depending upon the type of network being addressed and the various constraints it uses. As we use the hybrid authentication scheme in our scenario, the authentication is completed successfully before a device is provided access to the network.

Only after the initial authentication process is successfully completed, the new device is provided with a public key K_{PU} (Figure 6.2) to communicate with the network server using a secure line. Before the key-pairs K_P are generated, the new device also calculates a list of possible edge nodes that can be used to initiate the key exchange process. This list of edge devices N_E is used while generating encryption keys and is defined further in the next section.

6.3.2 Key-Pair Generation

The key generation process is usually an integral part of the scheme as the strength of the key is directly related to the amount of time it will take to break it. It also depends upon the type of devices being used in the network as the key length

will be selected based on the computation possible with onboard resources. As in a distributed network, the data travels using hops, the size of the encrypted data is also an important factor when selecting an encryption algorithm [148].

Our scheme uses hybrid encryption techniques which initially requires a set of asymmetric keys, so only the public key K_{PU} is exchanged which is unique for each device on the network. In addition, as the key pair K_P is generated by the device itself, the private key K_{PI} never leaves the host device and hence an adversary cannot predict the private key. However, unlike other key distribution schemes, which generate a single key-pair, our proposed scheme generates a pool of key-pairs K_{POOL}.

This pool of key-pairs K_{POOL} decreases the probability of finding the actual key, which is randomly selected from the pool by the server, amongst the rest of the dummy keys. Each key pair K_P generated uses a different version of the seed value S_V to avoid similarity in the generated pairs and will provide the same level of encryption. All key-pairs K_P are regarded as the same until the server picks one (the chosen key pair for the device) making the rest into dummy keys.

The number of keys generated K_{POOL} is directly proportional to the number of reachable edge devices N_E. As the key pool is divided amongst the available edge devices D_E for increased randomness, more key-pairs are generated if more edge devices are in range. Each edge node is given a subset of the pool of keys N_S generated to send to the server.

The number of keys also varies based on the maximum size of data each packet can contain and the key length. On an average, 50 key-pairs are generated for each available edge device and only the public keys are sent out in sets.

6.3.3 Key Transmission

The scheme takes advantage of the availability of multiple nodes/points of connectivity to the network. Instead of sending all the keys through one point of contact, sub-sets of the key pool (key-set) N_S are created and distributed amongst the reachable nodes D_E (Figure 6.1) thus further increasing the anonymity in the key exchange process.

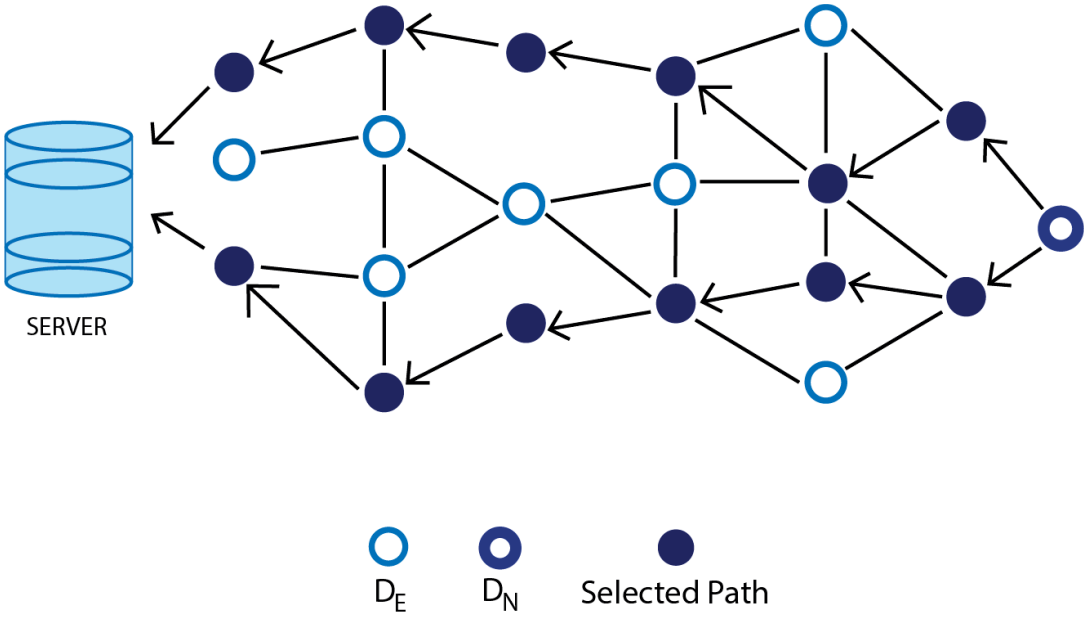

Figure 6.1 Multi-Path Data Transmission

The key set can be transmitted in both the static network by using a routing table with predefined paths, and the dynamic network by using a greedy algorithm and dynamically selecting the next hop nodes. The transmission's anonymity will be less for a predefined path as compared to that of a dynamic one provided by the GLOR protocol, but both will provide adequate anonymity to increase the overall security of the scheme.

The protocol also provides an additional protection against malicious nodes as the key pool is divided amongst the available edge nodes. This prevents the malicious node getting hold of all possible keys. In addition to multiple paths, the packets carrying the keys are encrypted and look like any normal data packet to make it harder to trace it across the network.

6.3.4 Key Selection

As the different sets of keys reach the server through different paths, it is important to know when they will expire. Hence, every set of keys sent across has a timeout value present in it (TTL). The server will check this value when it receives the packet and will accordingly discard any packets (with the key sets) whose TTL has expired.

The server will then randomly select a public key K_{PU} from the collective pool of all the keys received within the TTL. This key will then be registered as the communication key for the new device and can be requested by any device on the network to securely communicate with the new device.

The server will also allocate a public key from its personal pool of key pairs using which the new device can contact it. The server uses multiple key pairs for enabling devices on the network to communicate with it. The server, however, does not generate a unique key for each device, instead, it uses a key for a group of a predefined number of devices. Once the server's most recent public key is distributed to the pre-set number of devices, it creates a new key pair and will start allocating it to another set of new devices.

6.3.5 Challenge - Response

The server, upon having selected a public key, will send the new node a mathematical challenge. This is achieved by creating a moderately complex mathematical question which is then encrypted (along with the server's own public key) using the new device's selected public key. By doing so, the server can ensure the new device knows which public key has been chosen. The process is similar to how the Hybrid Authentication scheme uses it for user verification.

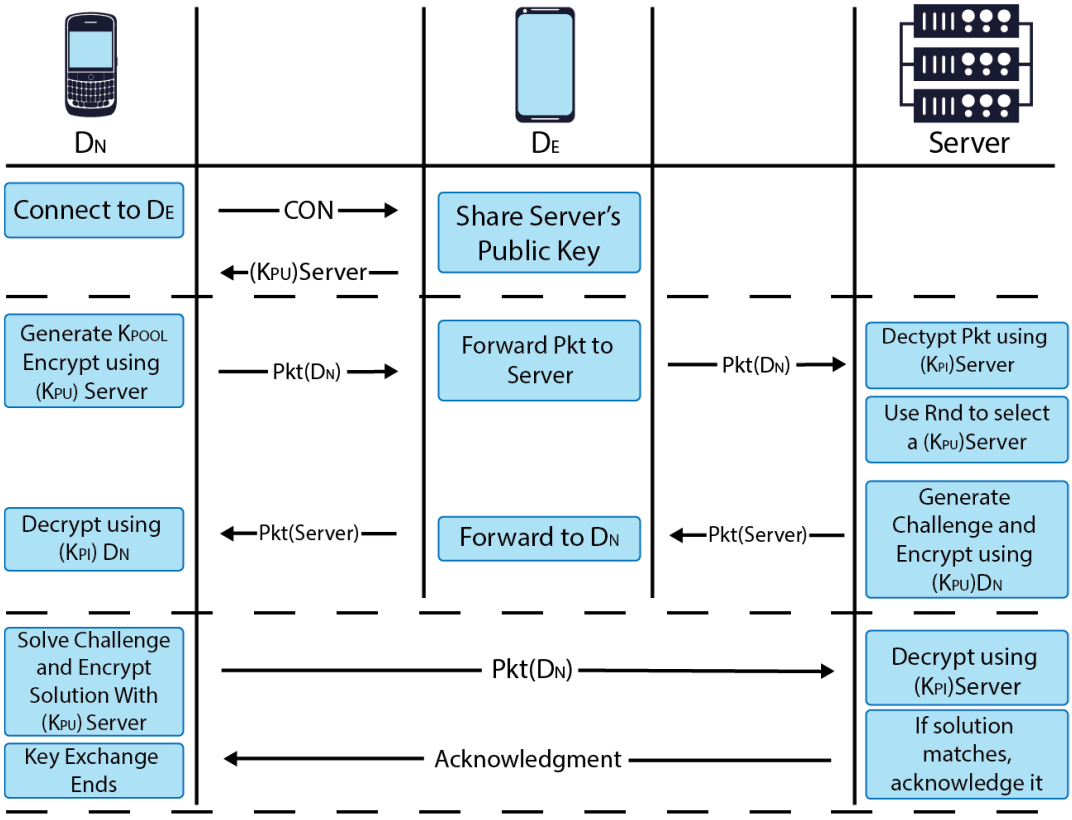

Figure 6.2 MPARK Distribution at a Glance

The new device upon receiving the challenge decrypts it with its pool of private keys. Once the new device can decrypt the challenge it knows which public key has been selected by the server and linked to the new device. It will then solve the challenge and encrypt the answer using the server provided public key. This is then

sent over to the server for confirmation that the new device knows the selected key and is now ready to become a part of the network.

6.4 Key Distribution Process

This section discusses the algorithm involved in the key distribution process. The process has been divided into various steps outlined by the MPARK distribution scheme. Refer to Table II. for details regarding the notations used in this section.

Once D_N is authenticated, it will establish an open line connection to any D_E in range. The connection at present is defined as an open line because no keys have been exchanged to enable encryption.

$$D_N\,(CON) \rightarrow D_E \in N_E \qquad (1)$$

All the D_E will provide D_N with Server (K_{PU}). As the server maintains a set of K_{PU}'s, individual D_E may possess different K_{PU} (Server) which they use to communicate with the server.

$$\text{Server}\,(K_{PU}): D_E \rightarrow D_N \qquad (2)$$

D_N will calculate the number of CON it has with D_E (N_E), based on which it will calculate the size of K_{POOL} to be used. The size of K_{POOL} depends upon the number of CON with D_E. A single Pkt will be sent to each D_E containing N_S of K_{PU}.

$$\sum \text{CON}\,(D_E) \rightarrow N_E \qquad (3)$$

$$N_S = N_E \qquad (4)$$

D_N will use random data for the S_V value to create a set of N_P K_P.

$$\text{Rnd}\,(S_V) \rightarrow D_N\,(K_{PU} \cup K_{PI}) \qquad (5)$$

$$N_P: (D_N (K_{PU} \cup K_{PI})) = N_P: D_N (K_P) \quad (6)$$

$$N_P: D_N (K_P) = K_{POOL} \quad (7)$$

The generated K_{PU} will be divided into N_E sets and individually encrypted with the server K_{PU} provided by each D_E.

$$\sum^{N_E} \text{Server} (K_{PU}) \rightarrow \sum^{N_S} D_N (K_{PU}) = N_S (\text{Pkt}) \quad (8)$$

The encrypted N_S will be converted into packets and will be given a TTL value. Then they will be distributed to the D_E corresponding to the server K_{PU} they provided.

$$\text{TTL} \cup N_S (\text{Pkt}) = \text{Pkt} (N_E) \quad (9)$$

$$\text{Pkt} (N_E) \rightarrow D_E (N_E) \quad (10)$$

Once D_E receives the encrypted Pkt, it will be sent across the network to the server. Each Pkt may take a different time and a different path to reach the server.

$$\text{Pkt} (D_N): D_E \rightarrow \text{Server} \quad (11)$$

The server will collect the Pkt's with a valid TTL, any late Pkt will be discarded. The server will then decrypt the received Pkt with its K_{PI} linked to the respective K_{PU} used to encrypt the Pkt.

$$\text{Server} (K_{PI}) \rightarrow \text{Pkt} (D_N) = \sum^{N_S} D_N (K_{PU}) \quad (12)$$

Once all the N_S (D_N) that made it to the server have been collected and decrypted, the server will then use a Rnd to select a single D_N (K_{PU}) from the N_S.

$$\text{Rnd}: \sum^{N_S} D_N (K_{PU}) \rightarrow D_N (K_{PU}) \quad (13)$$

Once a K_{PU} is selected, the Server will then prepare a mathematical challenge for D_N. The Server will encrypt this Mathematical challenge with the D_N (K_{PU}) it has selected. It will also include the Server (K_{PU}) for D_N to respond using a secure channel.

$$D_N (K_{PU}): (4 * 8) \cup \text{Server} (K_{PU}) \rightarrow \text{Pkt (Server)} \quad (14)$$

The server will then send the challenge over to D_N for confirmation of $D_N (K_{PU})$ selection and verification.

$$\text{Pkt (Server)} \rightarrow D_N \quad (15)$$

Once D_N receives the Pkt from the Server, it will start the process of decrypting it using its K_{POOL} of K_{PI}. In doing so, D_N will find out which K_P has been selected by the server.

$$\sum^{K_{POOL}} D_N (K_{PI}) \cup \text{Pkt (Server)} \rightarrow D_N (K_P) \quad (16)$$

$$D_N (K_P) \rightarrow D_N (K_{PI} \cup K_{PU}) \quad (17)$$

As D_N now knows the K_P chosen, it will use the K_{PI} to decrypt the Pkt received from the server to obtain the challenge. In addition, it will also get the K_{PU} of the server.

$$D_N (K_{PI}) \rightarrow \text{Pkt (Server)} \rightarrow (4 * 8) \cup \text{Server} (K_{PU}) \quad (18)$$

The D_N will then solve the challenge and as before, it will encrypt the answer using Server (K_{PU}) it just received. D_N will also Include its K_{PU} to confirm its selection.

$$(4 * 8) = 32 \quad (19)$$

$$\text{Server} (K_{PU}) \rightarrow ((32) \cup D_N (K_{PU})) \rightarrow \text{Pkt} (D_N) \quad (20)$$

D_N will now send the Pkt to the server and finish the key exchange process from its end.

$$\text{Pkt} (D_N) \rightarrow \text{Server} \quad (21)$$

Once it receives the Pkt, the Server will use its K_{PI} linked to the K_{PU} provided to D_N to decrypt the Pkt.

$$\text{Server } (K_{PI}) \rightarrow \text{Pkt } (D_N) = (32) \cup D_N (K_{PU}) \quad (22)$$

The Server will check the answer received and if D_N's selected K_{PU} is the same as the one received. If all is well, the Server will also end the key exchange process from its side.

$$(32) = (4 * 8) \rightarrow \text{True} \quad (23)$$

$$\text{Server } (D_N (K_{PU})) = D_N (D_N (K_{PU})) \rightarrow \text{True} \quad (24)$$

6.5 Security Analysis

In this section, we discuss the claims brought forward by proposed MPARK distribution scheme and their proof. We will also analyse the strength of our proposed scheme against known security threats relevant to distributed networks.

6.5.1 Claims & Proof

The proposed key distribution scheme makes the following claims:

Claim 1: The proposed scheme introduces multi-key approach making it computationally complex for the intruder to discover the communication key being used.

Proof: According to the MPARK distribution scheme, a new device must create a pool of keys amongst which one of the keys is selected by the server as the communication key for the new device. As all the keys generated have the identical strength and are of the same length, it is computationally complex to determine which one would be selected as the communication key for the new device.

$$P_1 \text{ (Key Discovery)} = 1 / K_{POOL} \quad (25)$$

As we can see from the equation above, the probability P_1 of a key being found is inversely proportional to the number of keys generated as defined by the size of K_{POOL}.

Claim 2: The proposed scheme incorporates a multi-path approach to distribute keys to avoid potential internal threats.

Proof: In addition to using a pool of keys, public keys are organized into a set of keys which are then sent across the network to the server with the help of multiple edge devices. Each edge node uses a different path to transmit the key set.

The use of this organized set of public keys ensures the server at least has a sub-set of the K_{POOL} to pick a key from. The number of keys in each set is determined by the number of edge devices in range of the new device.

$$N_S = K_{POOL} / N_E \quad (26)$$

$$P_2 (\text{Key Discovery} \mid P_1) = P_1 / N_S \quad (27)$$

By distributing the generated keys into multiple devices, the probability P_2 of key discovery is further reduced based on the number of edge devices available. As seen in the equation above, P_2 is further reduced when P_1 is added to the equation. In addition, if the nature of the path is dynamic, we can consider additional randomness at each hop taken by each set of keys as an additional improvement over P_2.

Claim 3: By incorporating anonymity in the key exchange process the proposed scheme ensures data packet confidentiality by concealing source information.

Proof: The data packets used in the MPARK distribution scheme are disguised as a normal data packet. In addition to the disguise, the source information of the packet is retracted from the header and is replaced by the last hop information as shown by the GLOR protocol packet structure in Chapter 3.3.4. The source

information is added to the encrypted part of the message. This information is only intended for the server because only the server will be able to decrypt the packet.

This ensures that while the packet travels through the network, its source and data cannot be identified or tracked providing anonymity.

Claim 4: Use of challenge-response in the proposed scheme makes it computationally improbable to tamper with the selected key.

Proof: According to the MPARK distribution scheme, once the server has selected a key it does not relay it back to the new device but instead uses a challenge response. As the new device possesses all the private keys, the server creates a mathematical challenge, encrypts it with the selected key and sends it to the new device.

By using a challenge-response, the selected key does not need to be transmitted back, just the encrypted message. As only the new device can decrypt the message using its pool of private keys, it will know which public key was used by the server to encrypt the packet. It can then send the solution to the challenge provided by the server confirming the new device now knows which public key was selected and thereby ending the key exchange.

6.5.2 Threat Analysis

As devices in the distributed networks are not usually monitored, there are various types of attacks, some affect their vicinity whereas others may affect the whole network [76, 89]. Below, we discuss the major security threats involving the key distribution and how MPARK distribution scheme validates against them.

6.5.2.1 Rogue / Compromised Network Device

In the event of a network device being compromised or going rogue, there is the very certain risk of the network becoming unstable. The major risk is the use of symmetric keys or the same set of asymmetric keys for the entire network. The rogue / compromised device will have the capability to intercept and disrupt any exchange on the network whether it's open line or encrypted.

In our scheme, if a device (or multiple devices) is compromised, it would not affect the network. As described in Section 6.3.1, each device uses its unique key-pair to communicate with other devices and the server. In addition, only the public keys are ever distributed and the private key never leaves the host device ensuring that only the host device can decrypt the data destined for it. Hence, the only key pair compromised is of the compromised device itself and it would still be able to only access data destined for it.

6.5.2.2 Sink Hole Attack

This attack is caused due to a rogue / compromised device requesting all traffic to itself to tamper or redirect the data packets (also known as selective forwarding). This attack can also interfere with an ongoing key exchange and hinder the process. As defined in Section 6.3.3, our proposed scheme uses a multipath approach where the key exchange takes multiple paths across the network. Unless a major part of the network isn't compromised causing a massive sinkhole, at least a partial set of public keys, if not all, will reach the server.

6.5.2.3 Black Hole Attack

The attack occurs as a result of a device dropping all the packets that it receives from any nearby devices. The proposed scheme uses a similar approach towards black hole attack as it does for sinkhole attack by utilizing multiple paths for transmission.

6.5.2.4 (Distributed) Denial of Service Attack

A DOS or DDOS attack may not have that much impact on network devices as overcrowding a device would just result in the data taking a different, less congested path. A DOS / DDOS attack on the server can result in relative delays in the network by slowing down the requests. However, this can be overcome by the server using a detection system as it already has enough power and resources to run it. It can also be resolved if the multiple synced servers are used to balance the request loads.

6.5.2.5 Rogue Edge Device

In the scenario that the edge device involved in the authentication process turns out to be a rogue device, the key exchange may be compromised or tampered with. The MPARK distribution scheme uses multiple edge devices to initiate the key exchange process, as described in Section 6.3.3, to make sure that a subset of all the keys generated is successfully transferred to the server through other edge devices. In addition, the pool of keys being sent is encrypted with different server public keys making it highly improbable for the rogue edge device to decrypt it.

6.5.2.6 Acknowledgement Spoofing

A compromised device can disrupt an ongoing key exchange with the server by sending a spoofed acknowledgement message to the device or even the server. This attack can be used in addition to other attacks to avoid the attack being detected by neighbour devices or the server.

This is avoided using a mathematical challenge that can only be solved by the device whose public key was used to encrypt the challenge in the first place. As described in Section 6.3.5, the MPARK distribution scheme uses a mathematical challenge to acknowledge the receipt and selection of the public key for the new node thereby avoiding any spoofed acknowledgement.

6.5.2.7 Man In The Middle Attack

A MITM attack during the key distribution process is a major threat. It can result in tampering with the key, impersonation and even identity theft causing a major network failure. Therefore, an important part of the MPARK distribution scheme is to prevent such an event. This is achieved by using a combination of randomness in the key transmission and selection process along with the anonymity achieved by disguising packets and the use of dummy keys.

When combined, the multipath approach, random key generation and selection, anonymous key transmission, mathematical challenge and the use of dummy keys significantly decrease the probability of the selected key being tampered with.

6.6 Summary

The key distribution scheme is a vital component of the security framework of any distributed network. In this chapter, we proposed a new key distribution scheme designed for distributed networks. Its major components were discussed in detail along with the methodology and then analysed based on various threats.

The fact that MPARK distribution incorporates randomness, anonymity and uses a multi-path approach enables it to prevent well-known threats by minimizing the probability of compromise. With further development and use in combination with dynamic schemes like hybrid authentication and hybrid encryption, the MPARK distribution scheme will create a better security framework improving the security and strength of the Secure-GLOR network model.

Chapter 7

Conclusion and Future Direction

This thesis presented four major components required to develop, enhance, expand and secure the dynamic wireless mesh networks. These components have been discussed, implemented and tested in the last four chapters (Chapter 3 to Chapter 6) of this thesis. The dynamic wireless mesh network is an essential addition to the much larger future communication network. With features like self-healing and self-management along with the ease of setup and deployment, the dynamic wireless mesh network is a solution to various scenarios which a traditional network cannot address. In this chapter, we conclude our thesis by shedding light on the major contributions made through each chapter. Finally, we present possible future works based on our research.

7.1 Thesis Summary and Conclusions

The requirements for future communication requires many advancements over existing networks to make them more flexible and dynamic to support the challenges put forth by the next generation of connected devices. The dynamic wireless mesh network is a major contender for a future communication system due to its flexible properties along with sustainability, scalability and security.

In this thesis, we take forward this idea by proposing a new type of network model that comprises various modules suited for dynamic wireless mesh networks. The network model provides multiple new features along with addressing various concerns originating from traditional network models. A compatible security framework also works in tandem with the network model to address authenticity, integrity and confidentiality of the data. Below is a summation of the proposed work presented in each chapter.

The background study and related works relevant to the research area were covered in Chapter 2. This included a discussion regarding relevant network models, routing techniques and protocols. The discussion focused on the structure, working, implementation and the security implications.

Chapter 2 also provided an overview of the prevalent security threats and concerns faced by mesh networks alongside key security aspects that can be used to prevent such security threats. Based on the literature available, we identified the two key research problems; routing and security. These key research problems are the main reasons that drives the research in the rest of the chapters.

In Chapter 3 we presented our proposed Geo-Location Oriented Routing (GLOR) protocol, which is designed specifically to address the limitations of a traditional mesh routing protocol. The chapter discussed in detail the various

components of the routing protocol and how they work together. These components included a new addressing scheme based on the geological location model, a new type of data transmission technique using smart packets and a web register that aids the network's working and helps in scaling it.

The chapter also provided a brief overview of the proposed network model along with the security framework and a performance evaluation of the proposed network model. The proposed GLOR protocol provided a sustainable and evolving network model to enable wireless mesh network to expand further. The new features greatly complimented the structure and natural workings of the network enhancing its potential.

The main contributions made in Chapter 4 address the first component of the security framework, the authentication process. In this chapter, we presented a hybrid authentication scheme modelled around the dynamic nature of the mesh network topology. The presented hybrid authentication model addresses an authentication request depending upon the current connectivity, available devices and environmental constraints of the network. The chapter also presented a performance assessment for the amount of resources required and the time taken for the authentication accomplishment in different scenarios.

The Hybrid authentication scheme makes the authentication process adaptable to the constantly changing network. Using the hybrid authentication will enable the network to maintain high levels of security, even in the absence of a central authority.

Chapter 5 presents our proposed hybrid encryption technique, another important component of the proposed security framework. In the chapter, we discuss the impact of encryption over mesh networks and how we can balance the overhead, size, time and resource consumption to provide good security without having a major delay in the network. To do so, we tested different algorithms for

symmetric and asymmetric encryption techniques on our proposed network model to get some performance results and resource consumption. The chapter also provides a comparison between single encryption technique and hybrid encryption technique.

It was deduced from the extensive testing that a combination of asymmetric and symmetric encryption techniques provides the best performance results as in incorporates the best of both; the ease-of-use of the asymmetric encryption's public-private key pair and the fast speeds of symmetric encryption.

Another important component of the security framework, the MPARK key management scheme is presented in Chapter 6. The MPARK scheme ties together the network's security framework by providing a means of generating, distributing and storing the crypto keys that are used by both the hybrid authentication scheme and the hybrid encryption scheme.

The chapter focuses on reducing the probability of tampering with the key exchange by using anonymity, randomness and decoys while exchanging crypto keys. It also discusses how the scheme works on a mesh topology along with a theoretical analysis of its effectiveness against known security threats. The proposed MPARK distribution scheme, upon deployment on a larger scale network, would be able to provide a much safer way of exchanging encryption keys for new devices. The scheme also benefits from the size of the network and a larger size has the potential to further increase the odds of discovering/tampering the encryption keys during the exchange.

7.2 Future Work

Based on the work presented in this thesis, our main objective is to design and develop a new network model for the wireless mesh network. As we address the major components of the network model, there are still other smaller components that require investigation. The possible areas where the research can be directed in the future are as below.

The proposed GLOR protocol can be further refined by small performance improvements to the node discovery and registration processes. This involves designing a streamlined process for new devices separate from the general network traffic and a setup of an initial direct connection to the web register to accelerate the registration process.

The Web Register, as discussed in Chapter 3, can be improved upon by adding extended functionality and offline support. One way of achieving it is to use multiple instances placed at geological positions with maximum communication traffic. This will enable quicker responses resulting in lowered delays in data transmission. The web register can also be upgraded to use AI and observe patterns of requests, updates and network traffic. This data can be analysed to provide better response timings and to detect possible security threats in the network.

The routing technique can also benefit from using data compression techniques to reduce the size of data packets travelling across the network. This can help reduce the overhead, shorten transmission times and consume fewer resources. However, the data compression technique required for the network must work in tandem with the data encryption and decryption techniques to avoid data corruption.

Improvement in the communication hardware used can also help redefine the structure of the routing protocols. As the network model uses geo-location as a reference to a device, the communication hardware chipsets can be modified to incorporate the same to avoid conversions and placeholders used currently.

The Secure-GLOR security framework will also require regular additions to better protect the network for currently known threats and any future threats that the network model may encounter. Some aspects of the security model, such as the encryption process can also be refined over time to provide better performance with less resource consumption.

Further, various other improvements can be made by implementing and testing the network model in real-world scenario to obtain better performance results. These real-world results will help further development of the routing protocol and the security model extensively.

Bibliography

[1] P. P. Parikh, M. G. Kanabar, and T. S. Sidhu, "Opportunities and challenges of wireless communication technologies for smart grid applications," in *Power and Energy Society General Meeting, 2010 IEEE*, 2010, pp. 1-7: IEEE.

[2] Y.-J. Lin, H. A. Latchman, M. Lee, and S. Katar, "A power line communication network infrastructure for the smart home," *IEEE wireless communications,* vol. 9, no. 6, pp. 104-111, 2002.

[3] M. Sugano, T. Kawazoe, Y. Ohta, and M. Murata, "Indoor Localization System using RSSI Measurement of Wireless Sensor Network based on ZigBee Standard," in *Wireless and Optical Communications*, 2006, pp. 1-6.

[4] S. Ferdoush and X. Li, "Wireless sensor network system design using Raspberry Pi and Arduino for environmental monitoring applications," *Procedia Computer Science,* vol. 34, pp. 103-110, 2014.

[5] K. Maine, C. Devieux, and P. Swan, "Overview of IRIDIUM satellite network," in *WESCON/'95. Conference record.'Microelectronics Communications Technology Producing Quality Products Mobile and Portable Power Emerging Technologies'*, 1995, p. 483: IEEE.

[6] C. E. Fossa, R. A. Raines, G. H. Gunsch, and M. A. Temple, "An overview of the IRIDIUM (R) low Earth orbit (LEO) satellite system," in *Aerospace and Electronics Conference, 1998. NAECON 1998. Proceedings of the IEEE 1998 National*, 1998, pp. 152-159: IEEE.

[7] R. R. Mohassel, A. Fung, F. Mohammadi, and K. Raahemifar, "A survey on advanced metering infrastructure," *International Journal of Electrical Power & Energy Systems,* vol. 63, pp. 473-484, 2014.

[8] C. Zhen-Wei Qiang, "Broadband infrastructure investment in stimulus packages: Relevance for developing countries," *info,* vol. 12, no. 2, pp. 41-56, 2010.

[9] R. Bruno, M. Conti, and E. Gregori, "Mesh networks: commodity multihop ad hoc networks," *IEEE communications magazine,* vol. 43, no. 3, pp. 123-131, 2005.

[10] D. Chen, M. Nixon, and A. Mok, "Why wirelesshart," in *WirelessHART™*: Springer, 2010, pp. 195-199.

[11] P. Gardner-Stephen, S. Farouque, M. Lloyd, A. Bate, and A. Cullen, "Piloting the serval mesh and serval mesh extender 2.0 in vanuatu: Preliminary results," in *Global Humanitarian Technology Conference (GHTC), 2017 IEEE*, 2017, pp. 1-10: IEEE.

[12] N. Wang, N. Zhang, and M. Wang, "Wireless sensors in agriculture and food industry—Recent development and future perspective," *Computers and electronics in agriculture,* vol. 50, no. 1, pp. 1-14, 2006.

[13] D. Puthal and B. Sahoo, "Secure Data Collection & Critical Data Transmission in Mobile Sink WSN: Secure and Energy efficient data collection technique," pp. 1-76, 2012.

[14] P. D. Pradeep and B. A. Kumar, "A survey of emergency communication network architectures," *International Journal of u-and e-Service, Science and Technology,* vol. 8, no. 4, pp. 61-68, 2015.

[15] P. Lieser, F. Alvarez, P. Gardner-Stephen, M. Hollick, and D. Boehnstedt, "Architecture for Responsive Emergency Communications Networks," in *Global Humanitarian Technology Conference (GHTC)(accepted for publication). IEEE*, 2017, pp. 1-9.

[16] D. Puthal, S. Nepal, R. Ranjan, and J. Chen, "A secure big data stream analytics framework for disaster management on the cloud," in *High Performance Computing and Communications; IEEE 14th International Conference on Smart City; IEEE 2nd International Conference on Data Science and Systems (HPCC/SmartCity/DSS), 2016 IEEE 18th International Conference on*, 2016, pp. 1218-1225: IEEE.

[17] J. Rodrigues *et al.*, "Benchmarking wireless protocols for feasibility in supporting crowdsourced mobile computing," in *Distributed Applications and Interoperable Systems*, 2016, pp. 96-108: Springer.

[18] A. Nanda, P. Nanda, and X. He, "Geo-Location Oriented Routing Protocol for Smart Dynamic Mesh Network," in *High Performance Computing and Communications; IEEE 14th International Conference on Smart City; IEEE 2nd International Conference on Data Science and Systems (HPCC/SmartCity/DSS), 2016 IEEE 18th International Conference on*, 2016, pp. 891-898: IEEE.

[19] A. Nanda, P. Nanda, X. He, and A. Jamdagni, "A secure routing scheme for wireless mesh networks," in *International Conference on Information Systems Security*, 2016, pp. 393-408: Springer.

[20] A. Nanda, P. Nanda, X. He, A. Jamdagni, and D. Puthal, "A Novel Hybrid Authentication Model for Geo Location Oriented Routing in Dynamic Wireless Mesh Networks," in *Proceedings of the 51st Hawaii International Conference on System Sciences*, 2018, pp. 5532-5541.

[21] A. Nanda, P. Nanda, X. He, A. Jamdagni, and D. Puthal, "A hybrid encryption technique for Secure-GLOR: The adaptive secure routing protocol for dynamic wireless mesh networks," *Future Generation Computer Systems,* pp. 1-20, 2018.

[22] A. Nanda, P. Nanda, X. He, A. Jamdagni, and D. Puthal, "Secure-GLOR: An Adaptive Secure Routing Protocol for Dynamic Wireless Mesh Networks," in *Trustcom/BigDataSE/ICESS, 2017 IEEE*, 2017, pp. 269-276: IEEE.

[23] P. Wong, V. Varikota, D. Nguyen, and A. Abukmail, "Automatic android-based wireless mesh networks," *Informatica,* vol. 38, no. 4, p. 313, 2014.

[24] I. F. Akyildiz and X. Wang, "A survey on wireless mesh networks," *IEEE Communications magazine,* vol. 43, no. 9, pp. S23-S30, 2005.

[25] X. Wang, "Wireless mesh networks," *Journal of Telemedicine and Telecare,* vol. 14, no. 8, pp. 401-403, 2008.

[26] I. F. Akyildiz, X. Wang, and W. Wang, "Wireless mesh networks: a survey," *Computer networks,* vol. 47, no. 4, pp. 445-487, 2005.

[27] S. M. Faccin, C. Wijting, J. Kenckt, and A. Damle, "Mesh WLAN networks: concept and system design," *IEEE Wireless Communications,* vol. 13, no. 2, pp. 10-17, 2006.

[28] A. Raniwala and T.-c. Chiueh, "Architecture and algorithms for an IEEE 802.11-based multi-channel wireless mesh network," in *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*, 2005, vol. 3, pp. 2223-2234: IEEE.

[29] P. Jacquet and L. Viennot, "Overhead in mobile ad-hoc network protocols," INRIARR-3965, 2000.

[30] A. Raniwala, K. Gopalan, and T.-c. Chiueh, "Centralized channel assignment and routing algorithms for multi-channel wireless mesh networks," *ACM SIGMOBILE Mobile Computing and Communications Review,* vol. 8, no. 2, pp. 50-65, 2004.

[31] R. Draves, J. Padhye, and B. Zill, "Routing in multi-radio, multi-hop wireless mesh networks," in *Proceedings of the 10th annual international conference on Mobile computing and networking*, 2004, pp. 114-128: ACM.

[32] M. E. M. Campista *et al.*, "Routing metrics and protocols for wireless mesh networks," *IEEE network,* vol. 22, no. 1, pp. 6-12, 2008.

[33] M. G. Kaosar, H. M. Asif, T. R. Sheltami, and A. S. H. Mahmoud, "Simulation-based comparative study of on demand routing protocols for MANET," in *International Conference on Wireless Networking and Mobile Computing (ICWNMC'05), Chennai, India*, 2005, vol. 1, pp. 201-206.

[34] L. Abusalah, A. Khokhar, and M. Guizani, "A survey of secure mobile ad hoc routing protocols," *IEEE communications surveys & tutorials,* vol. 10, no. 4, pp. 78-93, 2008.

[35] S. Jain and K. Agrawal, "A Survey on Multicast Routing Protocols for Mobile Ad Hoc Networks," *International Journal of Computer Applications,* vol. 96, no. 14, pp. 78-91, 2014.

[36] T. Clausen and P. Jacquet, "Optimized link state routing protocol (OLSR)," RFC 3626, 2003.

[37] P. Jacquet, A. Laouiti, P. Minet, and L. Viennot, "Performance analysis of OLSR multipoint relay flooding in two ad hoc wireless network models," INRIARR-4260, 2001.

[38] P. Kuppusamy, K. Thirunavukkarasu, and B. Kalaavathi, "A study and comparison of OLSR, AODV and TORA routing protocols in ad hoc networks," in *Electronics Computer Technology (ICECT), 2011 3rd International Conference on*, 2011, vol. 5, pp. 143-147: IEEE.

[39] N. I. Sarkar and W. G. Lol, "A study of manet routing protocols: Joint node density, packet length and mobility," in *Computers and Communications (ISCC), 2010 IEEE Symposium on*, 2010, pp. 515-520: IEEE.

[40] T. Clausen, G. Hansen, L. Christensen, and G. Behrmann, "The optimized link state routing protocol, evaluation through experiments and simulation," in *IEEE Symposium on" Wireless Personal Mobile Communications*, 2001, pp. 1-6.

[41] B. Kannhavong, H. Nakayama, Y. Nemoto, N. Kato, and A. Jamalipour, "SA-OLSR: Security aware optimized link state routing for mobile ad hoc networks," in *Communications, 2008. ICC'08. IEEE International Conference on*, 2008, pp. 1464-1468: IEEE.

[42] C. Adjih, D. Raffo, and P. Muhlethaler, "Attacks against OLSR: Distributed key management for security," in *2nd OLSR Interop/Workshop, Palaiseau, France*, 2005, vol. 14, pp. 1-5.

[43] A. Hafslund, A. Tønnesen, R. B. Rotvik, J. Andersson, and Ø. Kure, "Secure Extension to the OLSR protocol," in *OLSR Interop and Workshop*, 2004, vol. 1004, pp. 1-4: Citeseer.

[44] F. Hong, L. Hong, and C. Fu, "Secure olsr," in *Advanced Information Networking and Applications, 2005. AINA 2005. 19th International Conference on*, 2005, vol. 1, pp. 713-718: IEEE.

[45] C. Adjih, T. Clausen, P. Jacquet, A. Laouiti, P. Muhlethaler, and D. Raffo, "Securing the OLSR protocol," in *Proceedings of Med-Hoc-Net*, 2003, pp. 25-27.

[46] E. Baccelli, P. Jacquet, D. Nguyen, and T. Clausen, "OSPF multipoint relay (MPR) extension for ad hoc networks," RFC 5449, 2009.

[47] A. Qayyum, L. Viennot, and A. Laouiti, "Multipoint relaying: An efficient technique for flooding in mobile wireless networks," INRIARR-3898, 2000.

[48] T. Clausen, C. Dearlove, P. Jacquet, and U. Herberg, "The optimized link state routing protocol version 2," RFC 7181, 2014.

[49] C. Perkins, E. Belding-Royer, and S. Das, "Ad hoc on-demand distance vector (AODV) routing," RFC 3561, 2003.

[50] I. D. Chakeres and E. M. Belding-Royer, "AODV routing protocol implementation design," in *Distributed Computing Systems Workshops, 2004. Proceedings. 24th International Conference on*, 2004, pp. 698-703: IEEE.

[51] O. Abedi, M. Fathy, and J. Taghiloo, "Enhancing AODV routing protocol using mobility parameters in VANET," in *Computer Systems and Applications, 2008. AICCSA 2008. IEEE/ACS International Conference on*, 2008, pp. 229-235: IEEE.

[52] E. M. Royer and C. E. Perkins, "An implementation study of the AODV routing protocol," in *Wireless Communications and Networking Confernce, 2000. WCNC. 2000 IEEE*, 2000, vol. 3, pp. 1003-1008: IEEE.

[53] C. Kim, E. Talipov, and B. Ahn, "A reverse AODV routing protocol in ad hoc mobile networks," in *International Conference on Embedded and Ubiquitous Computing*, 2006, pp. 522-531: Springer.

[54] A. K. Gupta, H. Sadawarti, and A. K. Verma, "Performance analysis of AODV, DSR & TORA routing protocols," *International Journal of Engineering and Technology,* vol. 2, no. 2, p. 226, 2010.

[55] M. V. Khiavi, S. Jamali, and S. J. Gudakahriz, "Performance comparison of AODV, DSDV, DSR and TORA routing protocols in MANETs," *International Research Journal of Applied and Basic Sciences,* vol. 3, no. 7, pp. 1429-1436, 2012.

[56] J. Broch, D. A. Maltz, D. B. Johnson, Y.-C. Hu, and J. Jetcheva, "A performance comparison of multi-hop wireless ad hoc network routing protocols," in

Proceedings of the 4th annual ACM/IEEE international conference on Mobile computing and networking, 1998, pp. 85-97: ACM.

[57] Z. J. Haas, M. R. Pearlman, and P. Samar, "The bordercast resolution protocol (BRP) for ad hoc networks," *IETF, MANET Internet Draft,* pp. 13801-14853, 2002.

[58] Z. J. Haas, M. R. Pearlman, and P. Samar, "The zone routing protocol (ZRP) for ad hoc networks," pp. 1-6, 2002.

[59] S. Giannoulis, C. Antonopoulos, E. Topalis, and S. Koubias, "ZRP versus DSR and TORA: A comprehensive survey on ZRP performance," in *Emerging Technologies and Factory Automation, 2005. ETFA 2005. 10th IEEE Conference on*, 2005, vol. 1, pp. 8 pp.-1024: Ieee.

[60] S. Nithya, G. A. Kumar, and P. Adhavan, "Destination-sequenced distance vector routing (DSDV) using clustering approach in mobile adhoc network," in *Radar, Communication and Computing (ICRCC), 2012 International Conference on*, 2012, pp. 319-323: IEEE.

[61] C. E. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers," in *ACM SIGCOMM computer communication review*, 1994, vol. 24, no. 4, pp. 234-244: ACM.

[62] D. B. Johnson and D. A. Maltz, "Dynamic source routing in ad hoc wireless networks," in *Mobile computing*: Springer, 1996, pp. 153-181.

[63] V. D. Park and M. S. Corson, "A highly adaptive distributed routing algorithm for mobile wireless networks," in *INFOCOM'97. Sixteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Driving the Information Revolution., Proceedings IEEE*, 1997, vol. 3, pp. 1405-1413: IEEE.

[64] J. Thomas, J. Robble, and N. Modly, "Off grid communications with android meshing the mobile world," in *Homeland Security (HST), 2012 IEEE Conference on Technologies for*, 2012, pp. 401-405: IEEE.

[65] A. A. Malik, A. Mahboob, and T. M. Khan, "Implementing MANET for Trustworthy Collaboration Using OSS and Android Based COTS Devices," in *Collaboration Technologies and Systems (CTS), 2016 International Conference on*, 2016, pp. 485-492: IEEE.

[66] T. H. Clausen, J. W. Dean, and C. Dearlove, "Mobile ad hoc network (manet) neighborhood discovery protocol (nhdp)," RFC 6130, 2011.

[67] J. Macker, "Mobile ad hoc networking (MANET): Routing protocol performance issues and evaluation considerations," RFC 2501, 1999.

[68] P. Gardner-Stephen, "The serval project: Practical wireless ad-hoc mobile telecommunications," in "Flinders University, Adelaide, South Australia, Tech. Rep," 2011.

[69] P. Gardner-Stephen, A. Bettison, R. Challans, and J. Lakeman, "The rational behind the serval network layer for resilient communications," pp. 1680-1685, 2013.

[70] P. Gardner-Stephen, R. Challans, J. Lakeman, A. Bettison, D. Gardner-Stephen, and M. Lloyd, "The serval mesh: A platform for resilient communications in disaster & crisis," in *Global Humanitarian Technology Conference (GHTC), 2013 IEEE*, 2013, pp. 162-166: IEEE.

[71] P. Gardner-Stephen, J. Lakeman, R. Challans, C. Wallis, A. Stulman, and Y. Haddad, "Meshms: Ad hoc data transfer within mesh network," *International Journal of Communications, Network and System Sciences,* vol. 5, no. 08, p. 496, 2012.

[72] Y. Li, P. Hui, D. Jin, and S. Chen, "Delay-tolerant network protocol testing and evaluation," *IEEE Communications Magazine,* vol. 53, no. 1, pp. 258-266, 2015.

[73] Opengarden. *Fire Chat* [Online]. Available: https://opengarden.com

[74] M. Rogers, E. Saitta, and B. Tyers. *The Briar Project* [Online]. Available: https://code.briarproject.org

[75] P. Papadimitratos and Z. J. Haas, "Secure routing for mobile ad hoc networks," in *SCS Communication Networks and Distributed Systems Modeling and Simulation Conference (CNDS 2002)*, 2002, vol. 31, pp. 1-12: San Antonio, TX.

[76] C. Karlof and D. Wagner, "Secure routing in wireless sensor networks: Attacks and countermeasures," *Ad hoc networks,* vol. 1, no. 2-3, pp. 293-315, 2003.

[77] T. R. Andel and A. Yasinsac, "Surveying security analysis techniques in MANET routing protocols," *IEEE Communications Surveys & Tutorials,* vol. 9, no. 4, pp. 70-84, 2007.

[78] M. Aswal, P. Rawat, and T. Kumar, "Threats and vulnerabilities in wireless mesh networks," *International Journal of Recent Trends in Engineering,* vol. 2, no. 4, pp. 155-158, 2009.

[79] Y. Zhang and Y. Fang, "ARSA: An attack-resilient security architecture for multihop wireless mesh networks," *IEEE Journal on Selected areas in communications,* vol. 24, no. 10, pp. 1916-1928, 2006.

[80] Y. L. Sun, Z. Han, W. Yu, and K. R. Liu, "A trust evaluation framework in distributed networks: Vulnerability analysis and defense against attacks," in *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings*, 2006, pp. 1-13: IEEE.

[81] H. Deng, W. Li, and D. P. Agrawal, "Routing security in wireless ad hoc networks," *IEEE Communications magazine,* vol. 40, no. 10, pp. 70-75, 2002.

[82] M. G. Zapata and N. Asokan, "Securing ad hoc routing protocols," in *Proceedings of the 1st ACM workshop on Wireless security*, 2002, pp. 1-10: ACM.

[83] N. B. Salem and J.-P. Hubaux, "Securing wireless mesh networks," *IEEE Wireless Communications,* vol. 13, no. 2, pp. 50-55, 2006.

[84] C. Alcaraz and J. Lopez, "A security analysis for wireless sensor mesh networks in highly critical systems," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews),* vol. 40, no. 4, pp. 419-428, 2010.

[85] M. S. Siddiqui, "Security issues in wireless mesh networks," in *Multimedia and Ubiquitous Engineering, 2007. MUE'07. International Conference on*, 2007, pp. 717-722: IEEE.

[86] E. Stavrou and A. Pitsillides, "A survey on secure multipath routing protocols in WSNs," *Computer Networks,* vol. 54, no. 13, pp. 2215-2238, 2010.

[87] P. Yi, Y. Wu, F. Zou, and N. Liu, "A survey on security in wireless mesh networks," *IETE Technical Review,* vol. 27, no. 1, pp. 6-14, 2010.

[88] D. Puthal, S. P. Mohanty, P. Nanda, and U. Choppali, "Building Security Perimeters to Protect Network Systems Against Cyber Threats [Future Directions]," *IEEE Consumer Electronics Magazine,* vol. 6, no. 4, pp. 24-27, 2017.

[89] R. Roman, J. Lopez, and M. Mambo, "Mobile edge computing, fog et al.: A survey and analysis of security threats and challenges," *Future Generation Computer Systems,* vol. 78, pp. 680-698, 2018.

[90] D. Puthal, S. Nepal, R. Ranjan, and J. Chen, "Threats to networking cloud and edge datacenters in the internet of things," *IEEE Cloud Computing,* vol. 3, no. 3, pp. 64-71, 2016.

[91] Y. Sun, Z. Han, and K. R. Liu, "Defense of trust management vulnerabilities in distributed networks," *IEEE Communications Magazine,* vol. 46, no. 2, pp. 112-119, 2008.

[92] N. Singh, G. Chhabra, K. P. Singh, and H. Saini, "A secure authentication scheme in multi-operator domain (SAMD) for wireless mesh network," in *Proceedings of the International Conference on Data Engineering and Communication Technology*, 2017, pp. 343-357: Springer.

[93] P. Guo, J. Wang, X. H. Geng, C. S. Kim, and J.-U. Kim, "A Variable Threshold-Value Authentication Architecture for Wireless Mesh Networks," *Journal of Internet Technology,* vol. 15, no. 6, pp. 929-935, 2014.

[94] T. Gao, F. Peng, and N. Guo, "Anonymous authentication scheme based on identity-based proxy group signature for wireless mesh network," *EURASIP journal on wireless communications and networking,* vol. 2016, no. 1, p. 193, 2016.

[95] Y.-C. Hu, A. Perrig, and D. B. Johnson, "Ariadne: A secure on-demand routing protocol for ad hoc networks," *Wireless networks,* vol. 11, no. 1-2, pp. 21-38, 2005.

[96] K. Sanzgiri, B. Dahill, B. N. Levine, C. Shields, and E. M. Belding-Royer, "A secure routing protocol for ad hoc networks," in *Network Protocols, 2002. Proceedings. 10th IEEE International Conference on*, 2002, pp. 78-87: IEEE.

[97] H. Lin, J. Ma, J. Hu, and K. Yang, "PA-SHWMP: a privacy-aware secure hybrid wireless mesh protocol for IEEE 802.11 s wireless mesh networks," *EURASIP Journal on Wireless Communications and Networking,* vol. 2012, no. 1, p. 69, 2012.

[98] M. S. Islam, M. A. Hamid, and C. S. Hong, "SHWMP: a secure hybrid wireless mesh protocol for IEEE 802.11 s wireless mesh networks," in *Transactions on Computational Science VI*: Springer, 2009, pp. 95-114.

[99] Y.-M. Lai, P.-J. Cheng, C.-C. Lee, and C.-Y. Ku, "A new ticket-based authentication mechanism for fast handover in mesh network," *PloS one,* vol. 11, no. 5, p. e0155064, 2016.

[100] Y.-C. Hu, D. B. Johnson, and A. Perrig, "SEAD: Secure efficient distance vector routing for mobile wireless ad hoc networks," *Ad hoc networks,* vol. 1, no. 1, pp. 175-192, 2003.

[101] S. P. Singh and R. Maini, "Comparison of data encryption algorithms," *International Journal of Computer Science and Communication,* vol. 2, no. 1, pp. 125-127, 2011.

[102] S. Heron, "Advanced encryption standard (AES)," *Network Security,* vol. 2009, no. 12, pp. 8-12, 2009.

[103] *Announcing the advanced encryption standard (AES)*, 2001.

[104] H. Alanazi, B. B. Zaidan, A. A. Zaidan, H. A. Jalab, M. Shabbir, and Y. Al-Nabhani, "New comparative study between DES, 3DES and AES within nine factors," *arXiv preprint arXiv:1003.4085,* pp. 152-157, 2010.

[105] P. Hamalainen, M. Hannikainen, T. Hamalainen, and J. Saarinen, "Configurable hardware implementation of triple-DES encryption algorithm for wireless local area network," in *Acoustics, Speech, and Signal Processing, 2001. Proceedings.(ICASSP'01). 2001 IEEE International Conference on*, 2001, vol. 2, pp. 1221-1224: IEEE.

[106] T. Nie and T. Zhang, "A study of DES and Blowfish encryption algorithm," in *Tencon 2009-2009 IEEE Region 10 Conference*, 2009, pp. 1-4: IEEE.

[107] A. Mousa, "Data encryption performance based on Blowfish," in *ELMAR, 2005. 47th International Symposium*, 2005, pp. 131-134: IEEE.

[108] P. Zimmermann, "A proposed standard format for RSA cryptosystems," *Computer,* no. 9, pp. 21-34, 1986.

[109] U. Somani, K. Lakhani, and M. Mundra, "Implementing digital signature with RSA encryption algorithm to enhance the Data Security of cloud in Cloud Computing," in *Parallel Distributed and Grid Computing (PDGC), 2010 1st International Conference on*, 2010, pp. 211-216: IEEE.

[110] R. S. Dhakar, A. K. Gupta, and P. Sharma, "Modified RSA encryption algorithm (MREA)," in *Advanced Computing & Communication Technologies (ACCT), 2012 Second International Conference on*, 2012, pp. 426-429: IEEE.

[111] P. Fei, Q. Shui-Sheng, and L. Min, "A secure digital signature algorithm based on elliptic curve and chaotic mappings," *Circuits, Systems and Signal Processing,* vol. 24, no. 5, pp. 585-597, 2005.

[112] E.-O. Blaß and M. Zitterbart, "Towards Acceptable Public-Key Encryption in Sensor Networks," in *IWUC*, 2005, pp. 88-93.

[113] J. Kelsey, B. Schneier, D. Wagner, and C. Hall, "Cryptanalytic attacks on pseudorandom number generators," in *International Workshop on Fast Software Encryption*, 1998, pp. 168-188: Springer.

[114] C. S. Petrie and J. A. Connelly, "A noise-based IC random number generator for applications in cryptography," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications,* vol. 47, no. 5, pp. 615-621, 2000.

[115] F. Monrose, M. K. Reiter, Q. Li, and S. Wetzel, "Cryptographic key generation from voice," in *Security and Privacy, 2001. S&P 2001. Proceedings. 2001 IEEE Symposium on*, 2001, pp. 202-213: IEEE.

[116] A. Stefanov, N. Gisin, O. Guinnard, L. Guinnard, and H. Zbinden, "Optical quantum random number generator," *Journal of Modern Optics,* vol. 47, no. 4, pp. 595-598, 2000.

[117] D. Liu, P. Ning, and R. Li, "Establishing pairwise keys in distributed sensor networks," *ACM Transactions on Information and System Security (TISSEC),* vol. 8, no. 1, pp. 41-77, 2005.

[118] S. A. Camtepe and B. Yener, "Key distribution mechanisms for wireless sensor networks: a survey," *Rensselaer Polytechnic Institute, Troy, New York, Technical Report,* pp. 05-07, 2005.

[119] L. Eschenauer and V. D. Gligor, "A key-management scheme for distributed sensor networks," in *Proceedings of the 9th ACM Conference on Computer and Communications Security*, 2002, pp. 41-47: ACM.

[120] H. Chan, A. Perrig, and D. Song, "Random key predistribution schemes for sensor networks," in *Security and Privacy, 2003. Proceedings. 2003 Symposium on*, 2003, pp. 197-213: IEEE.

[121] B. Wu, J. Wu, E. B. Fernandez, M. Ilyas, and S. Magliveras, "Secure and efficient key management in mobile ad hoc networks," *Journal of Network and Computer Applications,* vol. 30, no. 3, pp. 937-954, 2007.

[122] T.-H. Wu, "A passive protected self-healing mesh network architecture and applications," *IEEE/ACM Transactions on Networking (TON),* vol. 2, no. 1, pp. 40-52, 1994.

[123] C. Dearlove and T. Clausen, "An optimization for the mobile ad hoc network (MANET) neighborhood discovery protocol (NHDP)," RFC 7466, 2015.

[124] T. Camp, J. Boleng, and V. Davies, "A survey of mobility models for ad hoc network research," *Wireless communications and mobile computing,* vol. 2, no. 5, pp. 483-502, 2002.

[125] H.-C. Jang, "Applications of Geometric Algorithms to Reduce Interference in Wireless Mesh Network," *arXiv preprint arXiv:1003.3569,* pp. 62-85, 2010.

[126] I. Stojmenovic, "Position-based routing in ad hoc networks," *IEEE communications magazine,* vol. 40, no. 7, pp. 128-134, 2002.

[127] P. Bose, P. Morin, I. Stojmenović, and J. Urrutia, "Routing with guaranteed delivery in ad hoc wireless networks," *Wireless networks,* vol. 7, no. 6, pp. 609-616, 2001.

[128] P. Gallagher, "Digital signature standard (dss)," *Federal Information Processing Standards Publications, volume FIPS,* pp. 186-3, 2013.

[129] Y. Lee, H. Lee, G. Lee, H. Kim, and C. Jeong, "Design of hybrid authentication scheme and key distribution for mobile multi-hop relay in IEEE 802.16 j," in *Proceedings of the 2009 Euro American Conference on Telematics and Information Systems: New Opportunities to increase Digital Citizenship*, 2009, p. 12: ACM.

[130] P. Papadimitratos and Z. J. Haas, "Secure link state routing for mobile ad hoc networks," in *Applications and the Internet Workshops, 2003. Proceedings. 2003 Symposium on*, 2003, pp. 379-383: IEEE.

[131] P. Papadimitratos and Z. J. Haas, "Secure message transmission in mobile ad hoc networks," *Ad Hoc Networks,* vol. 1, no. 1, pp. 193-209, 2003.

[132] J. Sen, "Security and privacy issues in wireless mesh networks: A survey," in *Wireless networks and security*: Springer, 2013, pp. 189-272.

[133] S. Yi, P. Naldurg, and R. Kravets, "Security-aware ad hoc routing for wireless networks," in *Proceedings of the 2nd ACM international symposium on Mobile ad hoc networking & computing*, 2001, pp. 299-302: ACM.

[134] X. Li, J. Niu, S. Kumari, F. Wu, A. K. Sangaiah, and K.-K. R. Choo, "A three-factor anonymous authentication scheme for wireless sensor networks in internet of things environments," *Journal of Network and Computer Applications,* vol. 103, pp. 194-204, 2018.

[135] F. Wu *et al.*, "An efficient authentication and key agreement scheme for multi-gateway wireless sensor networks in IoT deployment," *Journal of Network and Computer Applications,* vol. 89, pp. 72-85, 2017.

[136] Y. Yang, Z. Wei, Y. Zhang, H. Lu, K.-K. R. Choo, and H. Cai, "V2X security: A case study of anonymous authentication," *Pervasive and Mobile Computing,* vol. 41, pp. 259-269, 2017.

[137] A. Rasheed, A. Kenneth, R. Mahapatra, and D. Puthal, "Private matching and set intersection computation in multi-agent and industrial control systems," in *Proceedings of the 12th Annual Conference on Cyber and Information Security Research*, 2017, p. 14: ACM.

[138] D. Puthal, S. Nepal, R. Ranjan, and J. Chen, "DLSeF: A dynamic key-length-based efficient real-time security verification model for big data stream," *ACM Transactions on Embedded Computing Systems (TECS),* vol. 16, no. 2, p. 51, 2017.

[139] D. Puthal, X. Wu, S. Nepal, R. Ranjan, and J. Chen, "SEEN: A selective encryption method to ensure confidentiality for big sensing data streams," *IEEE Transactions on Big Data,* pp. 1-1, 2017.

[140] D. Puthal, S. Nepal, R. Ranjan, and J. Chen, "A synchronized shared key generation method for maintaining end-to-end security of big data streams," in *Proceedings of the 50th Hawaii International Conference on System Sciences*, 2017, pp. 6011-6020.

[141] O. Cheikhrouhou, M. Laurent-Maknavicius, and H. Chaouchi, "Security architecture in a multi-hop mesh network," in *Proc. 5th Conference on Security and Network Architectures (SAR 2006)*, 2006, pp. 1-10: Citeseer.

[142] A. Ramesh and A. Suruliandi, "Performance analysis of encryption algorithms for Information Security," in *Circuits, Power and Computing Technologies (ICCPCT), 2013 International Conference on*, 2013, pp. 840-844: IEEE.

[143] S. Zhu, S. Setia, and S. Jajodia, "LEAP+: Efficient security mechanisms for large-scale distributed sensor networks," *ACM Transactions on Sensor Networks (TOSN),* vol. 2, no. 4, pp. 500-528, 2006.

[144] D. K. Altop, M. A. Bingöl, A. Levi, and E. Savaş, "DKEM: Secure and efficient distributed key establishment protocol for wireless mesh networks," *Ad Hoc Networks,* vol. 54, pp. 53-68, 2017.

[145] S. Ruj, A. Nayak, and I. Stojmenovic, "Pairwise and triple key distribution in wireless sensor networks with applications," *IEEE Transactions on Computers,* vol. 62, no. 11, pp. 2224-2237, 2013.

[146] O. K. Sahingoz, "Large scale wireless sensor networks with multi-level dynamic key management scheme," *Journal of Systems Architecture,* vol. 59, no. 9, pp. 801-807, 2013.

[147] X. He, M. Niedermeier, and H. De Meer, "Dynamic key management in wireless sensor networks: A survey," *Journal of Network and Computer Applications,* vol. 36, no. 2, pp. 611-622, 2013.

[148] M. Abolhasan, B. Hagelstein, and J.-P. Wang, "Real-world performance of current proactive multi-hop mesh protocols," in *Communications, 2009. APCC 2009. 15th Asia-Pacific Conference on*, 2009, pp. 44-47: IEEE.