ABSTRACT


The increasing demand for ubiquitous Internet services imposes more security threats to communications due to open mediums in wireless networks. Thus, security mechanisms are proposed to protect communications, while putting more overheads on the transmission. As one of most widely used security mechanisms, authentication is used to identify mobile nodes (MNs), prevent unauthorized usage, and negotiate credentials with heavy overhead. Nevertheless, authentication mechanisms also induce heavy burdens, such as encryption/decryption load and long delay, in wireless networks. Although some solutions are proposed to reduce the burdens caused by the authentication, there have been little quantitative analysis, flexible protocol design, and optimized architecture implementation on the authentication that are adaptive to the quality of service (QoS) up to date.

In this thesis, we propose in-depth design and analysis of the authentication protocol and architecture to improve the authentication efficiency, such as delay and call dropping probability, in single- and multi-hop wireless networks. In the single-hop wireless networks, we first analyze the impact of authentication on the security and QoS quantitatively. Then, we present enhanced protocols for net-to-net and mobile-to-net authentication on hierarchical authentication architecture (HAA), which is the most widely used in wireless networks. The enhanced protocols are designed with the dynamic security associations (SAs) based on different functions of metrics to reduce the authentication delay and cost. Moreover, considering that HAA is not sufficient to network manageability and security, we further propose a new architecture with two control schemes for net-to-net and mobile-to-net authentication. The architecture is composed of licensed authentication centers and intelligent control schemes based on a utility function. The design of this architecture is effective to reducing the authentication latency, improving network scalability, and enhancing the network security in terms of reducing the number of SAs when inter-domain roaming happens.

In the multi-hop wireless networks, we propose reliable clustering algorithms to improve the service availability, which can cooperate with the proposed authentication protocols between clusters. In this design, the energy consumption and mobility of nodes
are evaluated quantitatively, and the proposed authentication protocols are entangled with the construction of hierarchical clusters dynamically, which is not only able to handle the failure of nodes efficiently, but also able to guarantee the security even from the start of constructing network architectures when mobile nodes frequently join and leave the multi-hop wireless networks.

As shown in the numerical and simulation results, by improving the authentication efficiency, such as delay and call dropping probability, in single- and multi-hop wireless networks significantly, our research demonstrates an in-depth impact of authentication on security and QoS in wireless networks, and builds a solid ground for future improvement of authentication protocols and architectures.
To my wife,

Ding, Hongping.

and my parents,

Liang, Wenjie
and
Zhang, Qiaoxiu.
Biography

Wei Liang (SM’04) received the B.S. degree from the Department of Electrical Engineering, Tsinghua University, Beijing, China, in 1998. Then, he went to Institute of Electronics, Chinese Academy of Sciences, Beijing, China, and received M.S. degree there in 2001. He is currently a Ph.D. candidate at North Carolina State University, Raleigh, NC. Since 2001, he has been a teaching assistant in the Department of Electrical and Computer Engineering, North Carolina State University. Then, he took part in the group of networking of wireless information systems (NetWIS) in 2002. His research interests include mobile and secure computing, authentication in wireless networks, quality of service in mobile networks, mobility management, modeling and performance analysis of wireless information networks.
Acknowledgements

I would like to express my gratitude to all those who gave me the possibility to complete this thesis. I want to thank the Department of Electrical and Computer Engineering (ECE) at North Carolina State University for giving me permission to do the necessary research work and support me during these years.

I am deeply indebted to my adviser Dr. Wenye Wang, who not only directed my work in these years, but also provided many good suggestions about the life. Her help, enlightened suggestions and encouragement motivated me in all the time of research and writing of this thesis. I also appreciate all the suggestions and helpful hints from professors, Dr. Michael Devetsikiotis, Dr. Khaled A. Harfoush, and Dr. Arne A. Nilsson, in my committee. Their proper doubts and questions on my research showed me a clear direction to complete my thesis successfully.

My former colleagues from the Department of ECE, especially my colleagues in the lab of Networking of Wireless Information Systems, supported me in my research work. I want to thank them for all their help, support, and valuable hints. I want to thank Avesh K. Agarwal, Nurcan Tezcan, Fei Xing, and Ming Zhao, for the helpful discussion.

Especially, I would like to give my special thanks to my wife, Hongping Ding, and my parents, Wenjie Liang and Qiaoxiu Zhang, whose patient love and strong support enabled me to pass the difficult time and complete this work.
## Contents

List of Figures viii  
List of Tables x  

1 **Introduction** 1  

2 **Authentication in Wireless Networks** 6  
  2.1 Introduction of Authentication 6  
  2.1.1 Secret Key Based Authentication 7  
  2.1.2 Public/Private Key Based Authentication 8  
  2.1.3 Security Association 9  
  2.2 Authentication in Single-Hop Wireless Networks 10  
  2.2.1 Authentication Architecture in Single-Hop Wireless Networks 10  
  2.2.2 Authentication Protocols in Single-Hop Wireless Networks 13  
  2.3 Authentication in Multi-Hop Wireless Networks 14  
  2.3.1 Resurrecting Duckling Policy for Secret-Key Based Authentication 15  
  2.3.2 Distributed Public-Key Based Authentication 15  

3 **Analysis of Authentication Mechanisms in Single-Hop Wireless Networks** 17  
  3.1 Effect of Authentication on Security and QoS 18  
  3.1.1 Overview of Challenge/Response Authentication 18  
  3.1.2 Effect of Authentication on Security and QoS Metrics 20  
  3.2 Quantitative Analysis of Challenge/Response Authentication 22  
  3.2.1 System Model and Metrics 22  
  3.2.2 Performance Metrics 24  
  3.2.3 Performance Analysis per Authentication 28  
  3.2.4 Arrival Rates of Authentication Requests 32  
  3.3 Numerical Results 36  
  3.3.1 Assumptions and Parameters 37  
  3.3.2 Effects of Mobility Pattern at Different Security Levels 38  
  3.4 Conclusions 41
4 Design of Authentication Mechanisms in Single-Hop Wireless Networks

4.1 Enhanced Authentication Protocols for Hierarchical Authentication Architecture

4.1.1 System Model and Metrics

4.1.2 Net-to-Net Authentication with Traffic-Based Security Association

4.1.3 Mobile-to-Net Authentication with Cost-Based Security Association

4.1.4 Mobile-to-Net Authentication with Risk-Aware Security Association

4.1.5 Numerical Results

4.2 Dynamic Authentication Architecture and Control Schemes

4.2.1 New Authentication Architecture

4.2.2 Net-to-Net Authentication with Traffic-Based Security Association

4.2.3 Mobile-to-Net Authentication with Threshold-Based Security Association

4.2.4 Simulation

4.3 Conclusions

5 Design and Analysis of Authentication Mechanisms in Multi-Hop Wireless Networks

5.1 System Model and Problem Formulation

5.1.1 System Model

5.1.2 Problem Formulation

5.2 Robust Clustering Algorithms to Improve the Service Availability

5.2.1 Evaluation of $MTTF_{c,r}(i)$

5.2.2 Adaptive Algorithms to Construct Hierarchical Clusters

5.3 Flexible Authentication Protocols on the Hierarchical Clusters

5.3.1 Management of Security Associations on Hierarchical Clusters

5.3.2 Authentication for Newly Joining Nodes

5.3.3 Authentication for Nodes Switching Clusters

5.3.4 Authentication for Nodes Initializing Communications

5.4 Simulation

5.4.1 Simulation Configuration

5.4.2 Performance Variation with Simulation Time

5.4.3 Performance Variation with Mobility

5.5 Conclusions

6 Conclusions and Future Directions

6.1 Summary of Thesis

6.1.1 Design and Analysis of Authentication in Single-Hop Wireless Networks

6.1.2 Design and Analysis of Authentication in Multi-Hop Wireless Networks

6.2 Future Directions
6.2.1 Future Research on the Authentication in Single-Hop Wireless Networks .................................................... 140

6.2.2 Future Research on the Authentication in Multi-Hop Wireless Networks .................................................... 141

Reference .......................................................................................................................... 141

Bibliography .................................................................................................................... 142
# List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Distributed Authentication Architecture</td>
<td>11</td>
</tr>
<tr>
<td>2.2</td>
<td>Hierarchical Authentication Architecture</td>
<td>13</td>
</tr>
<tr>
<td>2.3</td>
<td>SSL/TLS Four-Way Handshake Protocol</td>
<td>14</td>
</tr>
<tr>
<td>3.1</td>
<td>Challenge/Response Authentication in Public Wireless Access Networks</td>
<td>19</td>
</tr>
<tr>
<td>3.2</td>
<td>System Model of Authentication between Wireless Network Domains</td>
<td>22</td>
</tr>
<tr>
<td>3.3</td>
<td>Signaling Diagram at Security Level 1</td>
<td>25</td>
</tr>
<tr>
<td>3.4</td>
<td>Signaling Diagram at Security Level 2</td>
<td>26</td>
</tr>
<tr>
<td>3.5</td>
<td>Signaling Diagram at Security Levels 3 and 4</td>
<td>26</td>
</tr>
<tr>
<td>3.6</td>
<td>Time Diagrams of Events</td>
<td>33</td>
</tr>
<tr>
<td>3.7</td>
<td>Time Diagram for Number of Subnets Passed by in One Call</td>
<td>34</td>
</tr>
<tr>
<td>3.8</td>
<td>Authentication Cost vs. Residence Time in a Subnet</td>
<td>40</td>
</tr>
<tr>
<td>3.9</td>
<td>Authentication Time vs. Residence Time in a Subnet</td>
<td>40</td>
</tr>
<tr>
<td>3.10</td>
<td>Call Dropping Probability vs. Residence Time in a Subnet</td>
<td>41</td>
</tr>
<tr>
<td>3.11</td>
<td>Authentication Cost vs. Call Arrival Rate</td>
<td>42</td>
</tr>
<tr>
<td>3.12</td>
<td>Authentication Time vs. Call Arrival Rate</td>
<td>42</td>
</tr>
<tr>
<td>3.13</td>
<td>Call Dropping Probability vs. Security Levels</td>
<td>43</td>
</tr>
<tr>
<td>4.1</td>
<td>Hierarchical Authentication Architecture</td>
<td>46</td>
</tr>
<tr>
<td>4.2</td>
<td>Net-to-Net Authentication with Traffic-Based Security Association</td>
<td>47</td>
</tr>
<tr>
<td>4.3</td>
<td>Overview of Mobile-to-Net Authentication with Cost-Based Security Association</td>
<td>51</td>
</tr>
<tr>
<td>4.4</td>
<td>Authentication and Local SA Establishment Protocol</td>
<td>52</td>
</tr>
<tr>
<td>4.5</td>
<td>Demonstration of Chaining AAA Servers</td>
<td>53</td>
</tr>
<tr>
<td>4.6</td>
<td>Overview of Mobile-to-Net Authentication with Risk-Aware SA</td>
<td>56</td>
</tr>
<tr>
<td>4.7</td>
<td>Authentication Cost in Simulation</td>
<td>61</td>
</tr>
<tr>
<td>4.8</td>
<td>Authentication Cost vs. Arrival Rate of Authentication Requests</td>
<td>62</td>
</tr>
<tr>
<td>4.9</td>
<td>Authentication Cost vs. Increasing Number of Hops</td>
<td>62</td>
</tr>
<tr>
<td>4.10</td>
<td>Authentication Cost vs. Increasing Residence Time</td>
<td>63</td>
</tr>
<tr>
<td>4.11</td>
<td>Authentication Time vs. Residence Time of an MN</td>
<td>64</td>
</tr>
<tr>
<td>4.12</td>
<td>Authentication Cost vs. Number of Hops</td>
<td>64</td>
</tr>
<tr>
<td>4.13</td>
<td>Authentication Cost vs. Arrival Rate of Authentication Requests</td>
<td>65</td>
</tr>
</tbody>
</table>
## List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Comparison of Static and Dynamic Security Associations</td>
<td>9</td>
</tr>
<tr>
<td>3.1</td>
<td>Security Level Classification</td>
<td>25</td>
</tr>
<tr>
<td>3.2</td>
<td>Authentication Cost Symbols</td>
<td>28</td>
</tr>
<tr>
<td>3.3</td>
<td>Authentication Time Symbols</td>
<td>30</td>
</tr>
<tr>
<td>3.4</td>
<td>Parameters for Evaluation on QoS Metrics</td>
<td>37</td>
</tr>
<tr>
<td>4.1</td>
<td>Simulation Parameters for Different Networks.</td>
<td>60</td>
</tr>
<tr>
<td>4.2</td>
<td>Simulation Parameters of Costs for the Net-to-Net Authentication with</td>
<td>60</td>
</tr>
<tr>
<td></td>
<td>Traffic-Based SA.</td>
<td></td>
</tr>
<tr>
<td>4.3</td>
<td>Simulation Parameters for the Mobile-to-Net Authentication with Cost-</td>
<td>63</td>
</tr>
<tr>
<td></td>
<td>Based SA.</td>
<td></td>
</tr>
<tr>
<td>4.4</td>
<td>Simulation Parameters for the Mobile-to-Net Authentication with Risk-</td>
<td>66</td>
</tr>
<tr>
<td></td>
<td>Aware SA.</td>
<td></td>
</tr>
<tr>
<td>4.5</td>
<td>Definition of Symbols for the Mobile-to-Net Authentication on Proposed</td>
<td>77</td>
</tr>
<tr>
<td></td>
<td>New Architecture.</td>
<td></td>
</tr>
<tr>
<td>4.6</td>
<td>Simulation Parameters for the Mobile-to-Net Authentication on Proposed</td>
<td>87</td>
</tr>
<tr>
<td></td>
<td>New Architecture.</td>
<td></td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

The tremendous advance of wireless communication technologies has facilitated the ubiquitous Internet service, while inducing more challenges to security due to open medium [3]. In order to provide security services in wireless networks, authentication is used as an initial process to authorize a mobile node (MN) for communication through secret credentials [70]. In an authentication process, an MN is required to submit secret materials such as certificates or/challenge/response values for verification [14, 34, 37, 47, 66, 87]. The verification is performed by using a security association (SA), which is a relationship that affords security services with parameters such as session keys between the MN and its authenticator etc. With authentication process, the network resource can be maintained by authenticating legitimate users. The information secrecy and data integrity can also be guaranteed by using the negotiated secret credentials for encryption and message authentication. Therefore, the authentication service is directly related to the network security in terms of network resource, information secrecy, and data integrity.

Meanwhile, authentication also has great effects on the quality of service (QoS) in wireless networks. When certificate-based authentication mechanism, i.e. public/private-key based authentication mechanism, is applied, the computation complexity of encrypting/decrypting data with public/private keys consumes more time and power [28]. Therefore, secret key based authentication mechanisms, such as challenge/response authentication, are widely used in wireless networks [44, 72, 73], and are
able to reduce the cryptography load of authentication, while inducing other challenges in both single-hop and multi-hop wireless networks.

In single-hop wireless networks, such as cellular and Mobile IP networks, the challenges of authentication lie in obtaining the credentials, such as keys, for MN authentication when MNs are roaming among wireless networks. For intra-domain roaming MNs, i.e., the MNs who are moving in a homogeneous wireless system managed by a service provider, the problem to obtain the credentials, such as keys, for MNs has been solved by setting up a central authentication server for storing, verifying, and delivering the credentials. However, for inter-domain roaming MNs, i.e., the MNs who are moving among heterogeneous wireless systems managed by different service providers, the credentials of the MNs cannot be identified locally because local authentication server has no information about them. Although the local authentication server may have an indirect trust relationship with the server that stores the credentials for the MNs, the credentials of the MN are encrypted and transmitted for remote verification hop-by-hop between authentication servers, due to lack of end-to-end SA. The transmission and encryption/decryption of credentials affect many QoS parameters such as authentication cost in terms of signaling and encryption/decryption cost and authentication delay, which further affect other parameters such as call dropping probability.

In order to provide efficient and secure authentication in single-hop wireless networks, two major issues should be considered: authentication architecture and authentication scheme. The objective of authentication architectures is to provide secure inter-connection between wireless networks. To this end, the manageability of networks, which is measured by the number of SAs between networks, has been identified as a requirement in mobile environments [2]. The authentication scheme is designed to verify the user and generate credentials with mutual trust. Since the mutual trust is to protect the communication between networks and MNs, authentication process is necessary to provide security. On the other hand, efficiency of authentication with regard to authentication latency and bandwidth efficiency is also important, because an authentication process introduces an overhead of communications and radio links may be idle in authentication waiting time.

Similar as the authentication in single-hop wireless networks, most of the challenges of authentication in multi-hop wireless networks, such as Mobile Ad Hoc and sensor networks, are about acquiring the credentials, such as keys, of MNs for au-
This problem is more challenging in multi-hop wireless networks than in single-hop wireless networks due to the lack of infrastructures in multi-hop wireless networks. First, unlike the single-hop wireless networks with the support of servers, the MNs in multi-hop wireless networks usually cannot contact a server that stores credentials for thousands of MNs. Second, an MN has limited capability, e.g., small memory, to store the credentials for each MN that is communicating with itself. Thus, the authentication with secret-key in multi-hop wireless networks is difficult, and needs careful design and optimization. On the other hand, the impact of authentication on the system performance can be more important and challenging in multi-hop wireless networks because the energy, routing, and throughput will be affected greatly, which may further cause node failures due to the mobility and limited power of MNs, thus disrupting the communication any time.

In order to improve the authentication efficiency and security in multi-hop wireless networks, there is a trend to organize the MNs into clusters because it is consistent with the traditional management of wired networks if hierarchical architectures are applied, which enables the possible implementation of existing protocols in the multi-hop wireless networks. For example, by partitioning the MNs in the multi-hop wireless networks into hierarchical clusters, some hierarchical routing protocols in wired networks, such as Open Shortest Path First (OSPF), may be applied after revision [15]. In addition, the management of secret keys in the cluster groups without the support of central servers is easier and more scalable than the direct key management in flat multi-hop wireless networks [86].

The challenges of authentication mechanisms in both single-hop and multi-hop wireless networks have been investigated for many years [1, 9, 20, 24, 35, 44, 47, 49, 66, 70, 72, 73, 87]. However, current research of authentication mechanisms in the single-hop and multi-hop wireless networks is insufficient to solve the problems due to the following reasons. First of all, none of the research provides quantitative analysis of security and system performance, simultaneously, and nor do they show the connection between security and system performance clearly. Furthermore, mobility and traffic patterns are not considered, which are important features in wireless networks. Therefore, new authentication solutions may not be fully adapted to mobile environments with the concerns of security, performance, mobility and traffic patterns.

For single-hop wireless networks, current research on authentication architec-
ture is not sufficient to meet security and QoS requirements [2, 24, 25, 37, 44, 47, 72]. The authentication architecture is either based on a central authentication server, which is unrealistic for mass network environments [25], or based on chaining authentication servers with hop-by-hop static SAs between them, which is not suitable for large-scale and distributed networks [2, 24]. Furthermore, authentication with hop-by-hop SAs may induce security problems such as man-in-middle attack. A heavy burden of signaling cost and delay may be added to wireless networks by requiring hop-by-hop secure transmission for authentication. This overhead may degrade system performance such as bandwidth efficiency and call dropping probability.

As for multi-hop wireless networks, although clustering technique provides a flexible architecture for management in multi-hop wireless networks, the service availability, which includes the power, mobility, routing, and security evaluation, is not well considered due to the lack of integrated analysis of them on clusters [42, 56, 79]. First, they ignore the power status of each node, which provides fundamental basis of service availability. Second, they ignore the security, which may result in malicious fake of routing messages, furthermore compromising the data integrity and confidentiality in communications. Although some security protocols are proposed based on the constructed hierarchical clusters, they are not involved in the construction process, thus leaving a breach for attackers [10, 79].

Therefore, in this thesis, we propose the design and analysis of authentication mechanisms for both single-hop and multi-hop wireless networks. For single-hop wireless networks, in order to fill the blank field that there is no quantitative analysis of authentication impacts on the QoS and security simultaneously, we first analyze the authentication delay, cost, and call dropping probability on different security levels. Second, in order to improve the authentication efficiency in terms of delay and cost, we present enhanced authentication protocols for net-to-net and mobile-to-net authentication on hierarchical authentication architecture (HAA), which is the most widely used in single-hop wireless networks. Third, since we find that the manageability and security on the HAA are not well supported due to the use of hop-by-hop static SAs, we further propose a new architecture with two control schemes for net-to-net and mobile-to-net authentication. The new architecture is composed of licensed authentication centers (LACs), which can create and modify SAs on demand with intelligent control schemes. In the control schemes, more system parameters, such as bandwidth efficiency and num-
ber of SAs between networks, are taken into account than the proposed authentication schemes in the HAA. Thus, the manageability, security, and bandwidth efficiency are improved simultaneously.

For multi-hop wireless networks, in order to improve the effects of authentication on security and QoS in clusters, we propose reliable clustering algorithms as well as authentication schemes, which are entangled with the clustering algorithms, to guarantee the security even from the initial construction of clusters. In clustering algorithms, we quantitatively evaluate the service availability with the concern of power and mobility models. The hierarchical clusters then are built up based on the evaluation values, which provides a reliable seed bed for authentication service in multi-hop wireless networks. In the proposed authentication protocols, we handle the cases for joining, moving and communicating nodes. Thus, by integrating the improvements of power consumption, mobility, and flexible authentication, our reliable clustering algorithms and authentication protocols are able to improve the secure service availability in multi-hop wireless networks significantly.

The rest of this thesis is organized as follows. We introduce the background knowledge of authentication in wireless networks in Chapter II. In Chapter III, we analyze the authentication impacts on security and QoS in single-hop wireless networks with the example of challenge/response authentication. We further propose authentication protocols to improve the system performance in terms of authentication delay and cost on HAA in Chapter IV. Considering that the HAA cannot meet the requirement of network manageability and scalability very well, we design a novel distributed authentication architecture in Chapter V for the improvement of scalability and other system performance like bandwidth efficiency. In Chapter VI, we propose our reliable clustering algorithms and authentication protocols in multi-hop wireless networks, which cooperate together and provide a secure, reliable, and efficient environment for the communication in multi-hop wireless networks. Finally, we make conclusions about this thesis in Chapter VII.
Chapter 2

Authentication in Wireless Networks

In this chapter, we introduce the necessary knowledge to understand the authentication in wireless networks. We start from introducing the authentication mechanisms and some terminologies, such as security association. Then, the major issues and development of authentication in single-hop wireless networks are introduced. In third section, the current research and challenges on the authentication in multi-hop wireless networks are described.

2.1 Introduction of Authentication

The authentication in networks is defined as a process to identify a user with cryptography techniques. Based on the types of keys used for the authentication, the authentication mechanisms can be categorized into two types: secret-key based authentication and public/private-key based authentication.
2.1.1 Secret Key Based Authentication

The secret key based authentication is a process to identify a user by encrypting and decrypting the credentials with a secret key shared between communicators. In this type of authentication, a secret key, also called symmetric key, is pre-configured between two communicators. When an authentication process is initiated, the credentials, such as password or nonce, are encrypted and exchanged between the communicators with secret-key based cryptography techniques, such as DES. Depending on the type of materials exchanged for the authentication, the secret key based authentication mechanisms can be categorized into several types:

- **Password authentication**: the password authentication requires that a user inputs user name and password for the authentication. The user name and password are encrypted and transferred to a central server. The encrypted materials will be decrypted and verified at the central server [48].

- **Hash chain authentication**: the hash chain authentication requires that before a user applies for authentication, a chain of hash values need to be derived from a pre-shared value, e.g., secret key. When an authentication happens, a hash value derived at the end of the hash chain is used and removed after the successful authentication [63].

- **Challenge/response authentication**: the challenge/response authentication requires that an authenticator generates a challenge value, a random number, and transmits it to the user who needs authentication. The user encrypts the challenge value, and sends the result, called response value, back to the authenticator for verification [67, 73].

From the description above, we can see that the secret key based authentication mechanisms all depend the secret-key based cryptography techniques, such as DES. When applied in wireless networks, since the password authentication is not transparent to users, hash chain authentication and challenge/response authentication are all widely used in wireless networks.
2.1.2 Public/Private Key Based Authentication

The public/private key based authentication is a process to identify a user by using the uniqueness of public/private key set. In this type of authentication, a pair of asymmetric keys, public and private keys, are generated with a certain algorithms. A user who generates the pair of keys needs to transmit its public key to the authenticator before the authentication, and keep the private key for its own use. The private key matches the public key uniquely, and is computationally difficult to be derived from the public key. Only the data encrypted by the public key can be decrypted by the private key of the user. When a user needs to be authenticated, it encrypts a message with its private key, and sends it to its communication partner. The receiver is supposed to know the corresponding public key transmitted here previously. Because of the uniqueness of the public/private keys, the user can be identified.

In the public/private key system, in order to fulfill the uniqueness requirement for the public/private keys, complicated algorithms are developed to derive the key pair and encryption/decryption methods based on it. According to the algorithms and the whole procedure of the authentication, the classical public/private key based authentication mechanisms can be categorized into two types:

- Encryption-aided authentication: the encryption-aided authentication with public/private key works by encrypting each message with private key of a user. The process to identify a user depends on the unique matching of the private and public keys. The algorithms in this category include RSA and Ellipse Curve Cryptogram (ECC) [45, 76].

- Session-key-aimed authentication: the session-key-aimed authentication with public/private key works by exchanging public key with a user’s communication partner. The final purpose is to construct a session key with a user’s private key and the public key transmitted from its communication partner. The typical algorithms and the procedures of authentication in this category include Diffie-Hellman and its variants [76].

From the description above, we can see that the security of public/private key authentication relies on the unique match of the key pair, which is guaranteed by the complex algorithms to derive the keys. On the other hand, when using the public/private
key based authentication, either the encryption-aided authentication, or session-key-aided authentication needs to consume more energy and delay for calculation, which has been proved more than secret key based authentication [28].

2.1.3 Security Association

As defined in IP security architecture (IPsec), a security association (SA) is a one-way relationship between a sender and a receiver that afford security services to the traffic carried over communication sessions. An SA has many parameters, such as security parameters index (SPI), key and lifetime, all of which can be used for authentication [76]. Based on the definition of SA, we introduce two concepts, static SA and dynamic SA, which are defined and described as follows and in Table 2.1.

<table>
<thead>
<tr>
<th>Item</th>
<th>Static SA</th>
<th>Dynamic SA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lifetime</td>
<td>Long</td>
<td>Short</td>
</tr>
<tr>
<td>Number of SAs over time</td>
<td>More</td>
<td>Less</td>
</tr>
<tr>
<td>Manageability</td>
<td>Low</td>
<td>High</td>
</tr>
<tr>
<td>Security</td>
<td>Less</td>
<td>More</td>
</tr>
<tr>
<td>Establishment time</td>
<td>None</td>
<td>Long</td>
</tr>
<tr>
<td>Overhead of Communication</td>
<td>Less</td>
<td>More</td>
</tr>
</tbody>
</table>

- A dynamic SA is defined as an SA that is created on demand of communication session and exists for a short time to provide a temporary security service for the session. The existence of a dynamic SA can be adjusted by changing the lifetime of an SA. In other words, trust relationship between authentication parties can be removed when the lifetime is expired. Since dynamic SA is generated along with a communication session, applying dynamic SA can reduce the number of SAs, which in turn increases the manageability and security of networks by reducing the targets for attackers. However, the time to establish the dynamic SA is extensive [28], which may cause long authentication delay, further decreasing the bandwidth efficiency due to bandwidth idle during authentication waiting time.

- A static SA is defined as an SA that is independent with any communication sessions, and does not change for a long period of time, for example one month.
The meaning of ‘static’ is comparable to the ‘dynamic’ in terms of the relation with sessions, though static SAs can also be changed. The static SA can exist for a long time, which can reduce the time in establishing an SA for authentication, further improving the system bandwidth efficiency by reducing the link idle time. However, using static SAs increases the average number of SAs over time. A large number of SAs impose a great effort to manage the SAs and keep them safe across the networks, which causes manageability and security problems in wireless networks by exposing more targets to be attacked [2].

The protocols that can establish and modify SAs include Internet security association and key management protocol (ISAKMP), secure socket layer (SSL) and transport layer security (TLS). In these protocols, SSL and TLS are two protocols commonly used in wireless networks. SSL is a standard for encrypted client/server communication between network devices. TLS is an IETF standard with the goal to produce an Internet standard version of SSL [22]. A four-way handshake protocol in TLS allows the MN and the authentication server to negotiate encryption algorithms and exchange keys to set up an SA before any application data are transmitted. However, all of the algorithms applied in this protocol are time-consuming, especially when the client is an MN with limited calculation capability and power [28].

2.2 Authentication in Single-Hop Wireless Networks

As introduced in previous parts of this thesis, most of the authentication mechanisms in single-hop wireless networks are using secret-key based authentication. In order to deliver the necessary credentials for secret-key based authentication, two major issues, authentication architecture and protocol design, need to be solved.

2.2.1 Authentication Architecture in Single-Hop Wireless Networks

In order to deliver the credentials for authentication in single-hop wireless networks, many authentication architectures are proposed, which are dedicated to encrypting, transferring and verifying the credentials for MNs with static SAs [5, 11, 24, 43].
These solutions can be categorized into two types. One is distributed architecture; the other is and hierarchical architecture.

**Distributed Authentication Architecture**

In a distributed authentication architecture, an authentication center (AC) in a network shares static SAs with all ACs in different wireless networks [24]. An example of this distributed architecture is shown in Fig. 2.1. In this example, AC1, AC2, AC3 and AC4 are ACs in the corresponding networks 1, 2, 3 and 4. AC1, AC2, AC3 and AC4 are fully meshed with two SAs between each pair of them for two-way secure communications. This architecture provides end-to-end protection and guarantees the security between two networks. However, it may render the network unscalable because each AC must manage a huge amount of inter-domain SAs.

*Proposition 2.1:* Let $M_D$ be the number of ACs that need inter-domain SAs between each pair of them. The total number of inter-domain SAs in distributed authentication architecture, denoted as $N_D$, can be written as $N_D = M_D(M_D - 1)$.

We can see a quadratic growth in the number of SAs when the number of ACs increases. The appearance of one AC will cause $M_D$ SAs to be established. In hot spots such as personal area networks (PAN) and wireless local area networks (WLAN), networks may be setup and removed frequently. Therefore, the update of the SAs may cause a huge effort of management on this distributed architecture. Thus, the manageability of this architecture is limited, so it is not appropriate for dynamic mobile environments.
Hierarchical Authentication Architecture

In order to improve the manageability by reducing the number of SAs, a hierarchical authentication architecture is proposed in [24], where a proxy authentication center (PAC) is introduced to manage the SAs of a group of ACs. The PACs are organized in groups and are controlled by a higher-level PAC. Fig. 4.1 is an example of the hierarchical authentication architecture. In this example, AC1, AC2, AC3 and AC4 are ACs in the corresponding networks 1, 2, 3 and 4. AC1 and AC2 are controlled by PAC1, and AC3 and AC4 are managed by PAC2. The highest authentication proxy, PAC3, trusts two lower-level authentication proxies, PAC1 and PAC2. This architecture needs fewer inter-domain SAs, and the appearance of an AC only needs to inform one of the PACs, which greatly reduces the number of SAs between networks.

Proposition 2.2: Let $M_H$ be the number of ACs that need inter-domain SAs in hierarchical authentication architecture. The total number of inter-domain SAs, denoted as $N_H$, can be written as $N_H = \frac{2v}{v-1} (M_H - 1)$, where $v$ is the maximal number of ACs or PACs controlled by a PAC and $v > 1$.

Proof: Assume each PAC manages a number of $v$ ($v > 1$) ACs or PACs, the number of SAs will be

$$N_H = 2(M_H + \frac{M_H}{v} + \frac{M_H}{v^2} + \cdots + v)$$

$$= 2M_H \log_{\frac{1}{v}}^{M_H} \left( \frac{1}{v} \right)^i = \frac{2v}{v-1} (M_H - 1) \quad (2.1)$$

Because of this advantage, the hierarchical authentication architecture is deployed widely in wireless networks and proposed by IETF as a recommended standard for Mobile IP networks [24]. However, the hierarchical architecture does not provide end-to-end protection between two inter-domain ACs without direct trust relationship. This may cause many security problems, such as a man-in-middle or rogue proxy attack, in reality [2]. In addition, searching for the corresponding SA of a destination network along the hierarchical architecture may take a long time, even though two networks are adjacent, further deteriorating bandwidth efficiency of the system due to the idle
bandwidth during authentication waiting time. Moreover, the application of static SAs between ACs provides more time for potential intruders to attack.

To summarize, existing authentication architectures cannot meet the requirements of security and services in wireless networks due to large number of SAs and extensive waiting time for authentication. Moreover, these architectures do not support control policies on user level according to authentication traffic pattern. Therefore, we propose a new authentication architecture in Chapter 4.2, on which control schemes can be applied and the number of SAs can be reduced.

2.2.2 Authentication Protocols in Single-Hop Wireless Networks

There are many authentication protocols in single-hop wireless networks [1, 9, 22, 35, 44, 47, 49, 66, 72, 73, 87]. All of these protocols either focus on establishing an SA between two communication partners, or try to design efficient schemes to improve the authentication efficiency in terms of reduced signaling messages and cryptography load. As an example, we introduce an authentication protocol, four-way handshake protocol, in the standard of transport layer security (TLS) that is able to establish static SA between two communication partners.

A four-way handshake protocol in TLS allows the MN and the authentication server to negotiate encryption algorithms and exchange keys to set up an SA before any application data are transmitted.

The procedure of four-way handshake protocol is illustrated in Fig. 2.3 [28], which is divided into four phases. The first phase is to initiate a logical connection. At the second phase, a server sends a public key with its certificate to the client. The third
phase is performed by the client to provide its public key and certificate to the server after successfully receiving and verifying the public key of the server. The fourth phase is to confirm cipher specification parameters, such as keys, algorithms, and lifetime of the SA, based on the shared secrets. After these steps, an SA is established between the client and the server. However, all of the algorithms applied in this protocol are time-consuming, especially when the client is an MN with limited calculation capability and power [28]. In addition, the establishment of the SA cannot adapt to the QoS requirements of communications, e.g., the cost and delay of authentication cannot not be controlled and adjusted in the wireless environments.

Like the four-way handshake authentication protocol above, all of other authentication protocols in single-hop wireless networks have similar problem, i.e., they only consider the security and efficiency of authentication, and cannot adapt to the system performance.

2.3 Authentication in Multi-Hop Wireless Networks

The authentication in multi-hop wireless networks is very different from the authentication in single-hop wireless networks due to the lack of infrastructures and the weak capability of MNs. In order to deliver the necessary credentials for authentication, two types of authentication mechanisms are proposed: resurrecting duckling policy for authentication with secret keys [74, 75]; distributed public key based authentication [90].
2.3.1 Resurrecting Duckling Policy for Secret-Key Based Authentication

The resurrecting duckling policy was first proposed in [75]. The objective of this policy is to establish a secure transient SA between two devices, a master device and a slave device.

In this resurrecting duckling policy, the duckling is the slave device, and the mother duck is the master controller. The first entity that sends a secret key to a slave device through a secure channel will become the mother duck. This procedure is called imprinting. Then, the duckling can access other devices securely through its mother duck. The SA between the mother duck and the duckling will be broken in such events as SA life timeout. After that, a duckling can accept another imprinting. The SAs between devices can form a security chain, which corresponds to a tree topology of hierarchical master-slave relationships. This security model can be applied to very large ad-hoc networks, e.g. networks consisting of smart dust devices [12].

However, although the hierarchical relationships can be built through clustering techniques [?, ?] the service availability in those techniques are not well considered, which could make the SAs broken frequently, thus affecting the communications.

2.3.2 Distributed Public-Key Based Authentication

Public-key systems require a central trusted entity called certificated authority (CA), which is responsible to issue certificates by binding a public key to a node’s identity. However, this central entity decreases the scalability of ad hoc networks significantly, which drives the emergence of distributed public-key based authentication [90].

In [90], proposed method contributes to distributing trust to a set of nodes by having them sign certificates. The signature process is done using threshold cryptography [21]. An \((n; t + 1)\) threshold cryptography scheme allows \(n\) parties to share the ability to perform a cryptographic operation, so that any \(t + 1\) parties can perform this operation jointly, whereas it is infeasible for at most \(t\) parties to do so. Using this scheme, the private key \(k\) of the CA is divided into \(n\) shares \((s_1, s_2, \ldots, s_n)\), each share being assigned to each special node. Then, a set of \(t + 1\) special nodes is able to generate a valid certificate. As long as \(t\) or less special nodes are compromised and do not par-
nticipate in generating certificates, the service can operate. Even if compromised nodes deliver incorrect data, the service is able to sign certificates. Although this method is good at the network scalability, it consumes too much power because of the requirement to get signatures from $t$ MNs with public key cryptography.

On the other hand, self-organized public-key infrastructure is also proposed [32]. In this system, the CA is replaced by certificate chains. MNs issue certificates if they believe that a given public key belongs to given MNs. An MN stores list of certificates, which can be used to build a certificate chain. However, it has the similar problem we mentioned about the stability of clusters, i.e., the service availability is not well considered, thus increasing the possibility that SAs between MNs are broken.

From the fore-mentioned description about the distributed public-key based authentication, we can see that the authentication with public key in multi-hop wireless networks has its advantages and disadvantages. Although the certificates can increase the network scalability effectively, the energy consumption and stability in terms of usable service availability in clusters become important obstacles.
Chapter 3

Analysis of Authentication

Mechanisms in Single-Hop Wireless Networks

In this chapter, we quantitatively analyze the impacts of authentication on the security and system performance in single-hop wireless networks with the example of challenge/response, which is the first piece of work that builds up a direct numerical relationship between the security and QoS. In order to understand the analysis, we specify the effect of authentication on security and QoS in the first section. Then, we elaborate the analysis of authentication in the second section. We demonstrate the numerical results to show the impact of authentication on the security and QoS theoretically in the third section, and provide our conclusions in the fourth section of this chapter.
3.1 Effect of Authentication on Security and QoS

Challenge/response authentication is widely used in wireless networks, and it has significant effects on security and QoS in single-hop wireless network. Thus, we provide an overview of challenge/response authentication first. The effects of the authentication are described after that.

3.1.1 Overview of Challenge/Response Authentication

The authentication in wireless networks is defined as a process in which the MN needs to send out the secret credentials for verification and negotiate SAs for communications. An SA is a trust relationship with many parameters, such as keys and algorithms, for secure service with cryptographic techniques [76].

In a challenge/response-based authentication, a user is identified with shared SA by an authentication server that sends a challenge value, a random number, to the user for encryption, and verifies the returned value, called response value, with decryption [46]. In a foreign network, a visiting MN sends out an authentication request to an access point (AP), which is a function unit for transmitting data. The request is forwarded by the AP to a local authentication server (LAS), which only takes charge of authentication for the visiting MNs that are roaming from foreign networks. If the LAS has no enough information to verify the MN, it contacts the home authentication server (HAS) of the MN through an authentication architecture. An HAS is an authentication server that takes charge of the authentication for the MNs who subscribe the service in its network. And, an authentication architecture is composed of many authentication servers that share SAs with the LAS and HAS. Thus, when an HAS receives a request from the authentication architecture, it verifies the request by using an SA shared with the MN. If the request is an inter-domain authentication request, the HAS sends a registration request to the MN's home agent (HA), which is a router in the home network that maintains the current location of the MN, to update the MN's location.

Throughout this report, we assume that an MN is roaming in a foreign network domain. Then, the challenge/response authentication for an MN in a foreign network domain can be categorized into three types: intra-domain handoff authentication; session authentication; and inter-domain handoff authentication, with the signaling diagrams...
Intra-domain handoff authentication: When an MN crosses the boundary of subnets in the foreign network domain with an on-going service, an intra-domain handoff authentication is initiated. Since there is an on-going communication session between the MN and an AP, one session SA exists between the MN and the LAS in the visiting network domain. Therefore, it is unnecessary to contact the HAS of the MN for authentication. In the case shown in Fig. 3.1.A, the LAS who receives the authentication request from an MN sends a challenge value, i.e., a random value, to the MN. The MN encrypts the challenge value using shared SA with the LAS. The encrypted number, i.e., a response value, is replied to the LAS. After decrypting the replied value and comparing it with the original challenge value by the LAS, the LAS then can authenticate the MN.

Session authentication: When an MN starts a communication session in a subnet of a foreign network, a session authentication is initiated. Since there is no on-going communication session between the MN and the AP, session SA does not exist between the MN and the AP, and it is necessary to contact the HAS of the MN for authentication. In the case shown in Fig. 3.1.B, when the LAS receives the authentication request forwarded from the AP, it sends a challenge value to the MN. The MN encrypts the challenge value with the SA shared with the HAS, and replies the response value to the LAS. The LAS must forward the challenge and response values to the HAS of the MN for verification because the LAS does not share an SA with the visiting MN, and cannot decrypt the response value without the SA. After authentication at the HAS, the secret credentials such as keys to protect the communication may be generated and sent to the LAS.
Inter-domain handoff authentication: When an MN is crossing the boundaries of different foreign network domains with an on-going service, an inter-domain handoff authentication occurs. Since the session SA attached with the on-going communication session is between the MN and the other AP, no session SA exists between the MN and the new AP, and it is necessary to contact the HAS of the MN for authentication. In the case shown in Fig. 3.1.C, the signaling diagram is similar with that in the case of session authentication, except that the MN needs registration to its home agent (HA) through the HAS because we assume that the MN needs registration only if it is crossing the boundaries of different network domains.

3.1.2 Effect of Authentication on Security and QoS Metrics

Security services are to provide information secrecy, data integrity, and resource availability for users. Information secrecy means to prevent the improper disclosure of information in the communication, while data integrity is to prevent improper modification of data and resource availability is considered to preventing improper denial of services [76].

In order to provide security services in wireless networks, the challenge/response based authentication adopts several techniques to meet the requirements. First, the challenge/response authentication enables the MN to share an SA with its HAS. The SA is unique and secret to other users. Therefore, the identification of the MN is unique, which can prevent unauthorized MNs from accessing the network resource. Thus, the resource availability for authorized users can be guaranteed. Second, new secret credentials such as session keys are generated and sent to communication partners during authentication. The distributed secret credentials are used to encrypt the data of communication and provide message authentication code for data integrity check. Therefore, the authentication mechanism becomes a critical part to protect the information secrecy and data integrity because new secret credentials such as session keys are generated and transferred during this period. A well-designed authentication protocol then can provide great security by defeating well-known threats such as replay-attack and man-in-middle attack.

Besides the effect on the security, authentication also affects the QoS metrics, such as authentication delay, cost, call dropping probability and throughput of
communication due to the generation of the overhead of communications.

The authentication delay is defined as the time from when the MN sends out the authentication request to when the MN receives the authentication reply. During this authentication delay, no data for on-going service can be transmitted, which may interrupt the connections. Therefore, the call dropping probability is increased with the increase of authentication delay.

The authentication cost is defined as the signaling cost and processing load for cryptographic techniques. In a challenge/response authentication, the challenge/response values need to be transmitted back to the HAS of the MN for verification when the LAS has no SA shared with the roaming MN. Then, the signaling messages are transmitted between different LASs. The total number of signaling messages from the LAS to the HAS of the MN can be large if the authentication distance between them is long. Furthermore, the signaling messages need to be encrypted and decrypted hop-by-hop for protection due to lack of direct trust relationship between the LAS and the HAS. These multiple encryption and decryption increase the processing load of the networks. Moreover, the mobility and traffic patterns of MNs make the authentication happen frequently in different scenarios because the authentication is initiated when an MN starts a communication session or crosses boundaries of subnets with an on-going service, which may cause an imbalance distribution of authentication cost.

Compared to the effects of authentication on delay and cost, the throughput is affected by the authentication throughout the whole communication service. The throughput of the data communication is defined as the the effective data transmitted in a unit time. It can be greatly decreased due to authentication because of several reasons. First, when authentication happens, the authentication delay causes a temporary pause for data transmission, which decreases the throughput. Second, the cryptographic technique that is negotiated in authentication and protects the latter communication is changed from time to time during authentication, which depends on the mobility of the MN and different security requirements of the networks that the MN is visiting. The key size and algorithms to encrypt and decrypt the data affect the time to process the data. They will further reduce the effective data transmission rate due to the attachment of message authentication code for data integrity check.
3.2 Quantitative Analysis of Challenge/Response Authentication

In order to analyze the performance of challenge/response authentication in wireless networks, we discuss the effect of authentication on security and QoS based on challenge/response authentication first. Then, we describe a system model and define metrics used for performance evaluation in this section. We analyze these metrics at different security levels based on the mobility and traffic patterns, and provide the numerical results of our analysis on authentication cost, delay and call dropping probability [53, 54].

3.2.1 System Model and Metrics

The system model to describe the authentication interaction between interconnected wireless networks is shown in Fig. 3.2. In this model, there are a number of \( n \) autonomous wireless networks. Each network domain has a local authentication server (LAS) and a home authentication server (HAS). The figure shows the interaction between two networks, with APs (Access Points) and MNs (Mobile Nodes) moving between them.

![System Model of Authentication between Wireless Network Domains.](image)

**Figure 3.2:** System Model of Authentication between Wireless Network Domains.

of \( n \) autonomous wireless networks. Each network domain has a *local authentication*
server (LAS) and a home authentication server (HAS). The LAS and HAS are central authentication servers in a network domain. However, an LAS only takes charge of authentication for visiting MNs, while an HAS is only responsible for the authentication of the MNs that subscribe services in current network domain. The trust relationships between these LASs and HASs are maintained through an authentication architecture, which is an infrastructure composed of many proxy authentication servers and designed to securely deliver the authentication messages between authentication servers [24]. It is assumed that the LAS and HAS are integrated together, and the authentication architecture shares an SA with the LAS/HAS of a network domain.

Moreover, we need to describe the scenario and make assumptions on the mobility and traffic models.

**Scenario:** Assume that the challenge/response authentication is implemented on the generic system model with signaling diagrams shown in Fig. 3.1 because our initial assumption is that an MN is roaming in foreign network domains. For the intra-domain handoff authentication in foreign networks, Fig. 3.1.A shows the detailed process to realize this type of authentication. Similarly, the detailed processes on session and inter-domain handoff authentication in foreign networks are displayed in Fig. 3.1.B and Fig. 3.1.C, respectively.

**Mobility pattern:** The mobility pattern of an MN in our analysis is represented with the residence time of the MN in one subnet, denoted as $T_r$. We assume that $T_r$ is a random variable and the probability density function (PDF) of $T_r$, denoted as $f_{T_r}(t)$, is Gamma distribution with mean $1/\mu_r$ and variance $\nu$ [30]. Then, the Laplace transform of $f_{T_r}(t)$, $F_r(s)$, is

$$F_r(s) = \left(\frac{\mu_r \gamma}{s + \mu_r \gamma}\right)^\gamma, \quad \text{where} \quad \gamma = \frac{1}{\nu \mu_r^2}. \tag{3.1}$$

Furthermore, if the number of subnets passed by an MN is assumed to be uniformly distributed between $[1, M]$, the PDF of the residence time in a network domain, denoted as $f_{T_M}(t)$, can be expressed with a Laplace transform $F_M(s)$ as follows [83]:

$$F_M(s) = \frac{1}{M} \left(\frac{\mu_r \gamma}{s + \mu_r \gamma}\right)^\gamma \frac{1 - (\frac{\mu_r \gamma}{s + \mu_r \gamma})^\gamma}{1 - (\frac{\mu_r \gamma}{s + \mu_r \gamma})^\gamma}. \tag{3.2}$$

Then, the mean value of residence time in this network domain, denoted as $\overline{T_M}$, can be expressed as:

$$\overline{T_M} = -\frac{\partial F_M(s)}{\partial s} \bigg|_{s=0} = \frac{M + 1}{2\mu_r}. \tag{3.3}$$
Traffic pattern: In the analysis, we consider the call arrival rate and call duration time of the MN as the traffic patterns of the MN. First, we assume that the call arrival rate of the MN, which includes the incoming calls and outgoing calls, is Poisson process with average rate $\lambda_u$, then the PDF of the call inter-arrival time, denoted as $f_{T_A}(t)$, can be determined by:

$$f_{T_A}(t) = \lambda_u e^{-\lambda_u t}. \quad (3.4)$$

Moreover, we assume that a call duration time, denoted as $T_D$, has an exponential distribution with mean value $1/\eta$. Then, the PDF of call duration time, denoted as $f_{T_D}(t)$ can be written as:

$$f_{T_D}(t) = \eta e^{-\eta t}. \quad (3.5)$$

Based on these assumptions on the mobility and traffic patterns of the MN, we evaluate the security and QoS metrics of authentication when the MN is roaming in our generic system model. The security and QoS metrics needed for evaluation are defined in next section.

3.2.2 Performance Metrics

We categorize the performance metrics into security and QoS parameters. The security parameter is represented by security levels, at which different levels of protection are provided. Meanwhile, we consider authentication cost, delay and call dropping probability as the system performance for evaluation.

Security Levels

There are much quantitative analysis of QoS in networks [17, 18], whereas less analysis of security exists. This gap between the QoS and security analysis demands quantization of security for the engineering research. Therefore, the concept of security level becomes widely used for security evaluation [6, 69, 80]. However, all of them do not consider the nature of security, i.e., data integrity, secrecy, and availability. Therefore, we argue that the nature of security should become the standard to classify the security levels.

In our analysis, the security level is to indicate the level of protection provided by the authentication for quantitative analysis of security. The classification of
security levels is shown in Table 3.1 according to the security functions described in
Section 3.1.2, i.e., protection for integrity, secrecy and resource availability. Because of
different actions in challenge/response authentication, the protection of data integrity,
secrecy, and availability may be different at different security levels.

<table>
<thead>
<tr>
<th>Security Level $i$</th>
<th>Security Service</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Integrity</td>
</tr>
<tr>
<td>1</td>
<td>No</td>
</tr>
<tr>
<td>2</td>
<td>No</td>
</tr>
<tr>
<td>3</td>
<td>No</td>
</tr>
<tr>
<td>4</td>
<td>Yes</td>
</tr>
</tbody>
</table>

- **Security Level 1**: Any MNs can send data through an AP without authentication. The signaling diagram at this security level is shown in Fig.3.3.

![Signaling Diagram at Security Level 1](image)

- **Security Level 2**: Authentication is implemented with Media access control (MAC) address and no keys are generated for the subsequent communication.

- **Security Level 3**: Authentication is implemented with shared SA, and no keys are generated for the MN’s communication.

- **Security Level 4**: Authentication is implemented with shared SA, and keys are generated for data encryption and message integrity check.

**Average Authentication Cost**
In this context, we define authentication cost as the sum of signaling load and processing load for cryptographic techniques during one authentication operation. And, the average authentication cost, \( C(i) \), is defined as the sum of the authentication cost over a number of authentication requests in a unit time at security level \( i \), which can be written as:

\[
C(i) = \sum_{\beta=1}^{3} \lambda_\beta [C^{(s)}_\beta (i) + C^{(p)}_\beta (i)],
\]

(3.6)

where \( \beta \) is the index of authentication type. \( \beta = 1 \) represents an intra-domain handoff authentication, \( \beta = 2 \) means a session authentication, and \( \beta = 3 \) is an inter-domain handoff authentication. We denote \( C^{(s)}_\beta (i) \) and \( C^{(p)}_\beta (i) \) as the signaling load and processing load of cryptographic techniques, respectively, of an authentication with type \( \beta \).
at security level $i$. The arrival rate of requests for the authentication type $\beta$ is defined as $\lambda_\beta$, which is related with the mobility and traffic patterns of MNs.

**Average Authentication Delay**

We define *authentication delay* as the time from when the MN sends out an authentication request to when the MN receives the authentication reply. The *average authentication delay*, $T(i)$, is defined as the sum of an authentication delay over a number of authentication requests in a unit time at security level $i$. Then, $T(i)$ can be written as:

$$T(i) = \sum_{\beta=1}^{3} \lambda_\beta T_\beta(i),$$

where $T_\beta(i)$ is the authentication delay per operation at security level $i$ for authentication type $\beta$, and $\lambda_\beta$ is the arrival rate of authentication requests with type $\beta$.

**Average Call Dropping Probability during Authentication**

In order to consider the extended authentication delay and authentication failure in the definition of call dropping probability, the *call dropping probability* is defined as the probability that the service of an MN is dropped during one authentication operation because of either extended authentication delay [31, 84], or an authentication failure. When the MN roams among subnets in a network domain, the *average call dropping probability*, $P(i)$, is defined as the ratio of the sum of the call dropping probability per authentication in a unit time over the number of authentication requests sent by the MN within unit time at security level $i$. Let $P(i)$ denote the average call dropping probability at security level $i$, $P(i)$ can be written as:

$$P(i) = \frac{\sum_{\beta=1}^{3} \lambda_\beta [P_\beta(i) + P_e]}{\sum_{\beta=1}^{3} \lambda_\beta},$$

and

$$P_\beta(i) = P_{T_\beta(i)}(T_\beta(i) > T_{th}),$$

where $T_{th}$ is a threshold value of time, $P_{T_\beta(i)}(T_\beta(i) > T_{th})$ is the probability that an authentication delay is greater than the threshold time $T_{th}$ in authentication type $\beta$. $P_e$ is the probability that one authentication fails due to unknown damage on the credentials of a valid MN and it is unrelated with the security level $i$. Since there is no evidence on the pattern of attacks currently, we will use a mean value from experiments to represent $P_e$ in the numeric results of our analysis [65].

In summary, in order to evaluate $C(i)$, $T(i)$ and $P(i)$ in (3.6)~(3.8), we need
to analyze $\lambda_\beta$, $C^{(s)}_\beta(i)$, $C^{(p)}_\beta(i)$, $T_\beta(i)$, and $P_\beta(i)$. Next, we derive these parameters based on the system model, assumptions and the definitions of the performance metrics in Section 3.2.2.

### 3.2.3 Performance Analysis per Authentication

In this section, we analyze authentication cost, delay and call dropping probability for each authentication request. At different security levels, the authentication has different effects on the cost, delay and call dropping probability.

#### Authentication Cost per Operation

The authentication cost, $C_\beta(i)$, $(\beta = 1, 2, 3$ and $i = 1, 2, 3, 4)$, is composed of $C^{(s)}_\beta(i)$ and $C^{(p)}_\beta(i)$, which depend on the authentication type $\beta$ and security level $i$. For convenient analysis, we define a set of cost parameters in Table 3.2.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$c_s$</td>
<td>Transmission cost on one hop</td>
</tr>
<tr>
<td>$c_p$</td>
<td>Encryption/decryption cost on one hop</td>
</tr>
<tr>
<td>$c_v$</td>
<td>Verification cost at an authentication server</td>
</tr>
<tr>
<td>$c_{us}$</td>
<td>Encryption/decryption cost for a session key</td>
</tr>
<tr>
<td>$c_g$</td>
<td>Key generation cost</td>
</tr>
<tr>
<td>$c_{ts}$</td>
<td>Transmission cost for a session key to other communication identities</td>
</tr>
<tr>
<td>$c_{rg}$</td>
<td>Registration cost</td>
</tr>
</tbody>
</table>

Then, the transmission costs, $C^{(s)}_\beta(i)$, can be derived from the signaling diagrams in Fig. 3.3, 3.4, and 3.5, respectively, as follows:

$$C^{(s)}_\beta(i) = a_{\beta,i}c_s, \quad \forall/\beta = 1, 2, 3 \text{ and } i = 1, 2, 3, 4,$$

(3.9)

where $\beta$ is the authentication type, $i$ is the security level, and $a_{\beta,i}$ is an element of matrix $A$, which indicates the number of hops by which the whole authentication process passes for authentication type $\beta$ at security level $i$. $a_{\beta,i}$ can be obtained from observing the corresponding signaling figures. For example, when $\beta = 3$ and $i = 4$, $a_{3,4} = 2(N_h + 3)$ denotes the number of hops that the authentication signalings pass when $\beta = 3, \ i = 4$, which can be obtained from Fig. 3.5.B. Thus, we obtain $A$ as:
\[
A = \begin{bmatrix}
2 & 6 & 8 & 8 \\
2 & 2(N_h + 1) & 2(N_h + 2) & 2(N_h + 2) \\
2(N_h + 1) & 2(N_h + 2) & 2(N_h + 3) & 2(N_h + 3)
\end{bmatrix},
\]

(3.10)

where \( \beta \) and \( i \) represent the row and column of \( A \), respectively. \( N_h \) is the number of the hops between the MN and its HAS.

Similar with the analysis in (3.9), according to the signaling diagrams in Figs. 3.3, 3.4, and 3.5, \( C^{(p)}_{\beta}(i) \) can be written as:

\[
C^{(p)}_{\beta}(i) = \vec{b}_{\beta,i} \cdot \vec{x}_p, \quad \forall \beta = 1, 2, 3 \text{ and } i = 1, 2, 3, 4.
\]

(3.11)

Here, \( \vec{x}_p \) is a vector defined as:

\[
\vec{x}_p^T = [c_p, c_v, c_{us}, c_g, c_{ts}, c_{rg}],
\]

(3.12)

where all of the cost parameters are defined in Table 3.2. And, \( \vec{b}_{\beta,i} \) are also vectors determined by:

\[
\begin{align*}
\vec{b}_{1,1} &= \vec{b}_{2,1} = [0, 0, 0, 0, 0, 0], \\
\vec{b}_{1,2} &= [2, 1, 0, 0, 0, 0], \\
\vec{b}_{1,3} &= \vec{b}_{1,4} = [4, 1, 0, 0, 0, 0], \\
\vec{b}_{2,2} &= [2(N_h - 1), 1, 0, 0, 0, 0], \\
\vec{b}_{2,3} &= [2N_h, 1, 1, 0, 0, 0], \\
\vec{b}_{2,4} &= [2N_h, 1, 2, 1, 0, 0], \\
\vec{b}_{3,1} &= [0, 0, 0, 0, 0, 1], \\
\vec{b}_{3,2} &= [2N_h, 1, 0, 0, 0, 1], \\
\vec{b}_{3,3} &= [2(N_h + 1), 1, 1, 0, 0, 1], \\
\vec{b}_{3,4} &= [2(N_h + 1), 1, 2, 1, 1, 1].
\end{align*}
\]

(3.13)

**Delay per Authentication**

To derive the delay for different types of authentications in different security levels, we use the same signaling diagram shown in Fig. 3.1. We also define a set of time parameters shown in Table 3.3 for convenient description.

Then, \( T_{\beta}(i) \) can be expressed as:

\[
T_{\beta}(i) = \vec{d}_{\beta,i} \cdot \vec{x}_t, \quad \forall \beta = 1, 2, 3 \text{ and } i = 1, 2, 3, 4.
\]

(3.14)
Table 3.3: Authentication Time Symbols

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>(T_{pr})</td>
<td>Message propagation time on one hop</td>
</tr>
<tr>
<td>(T_{tr})</td>
<td>Message transmission time on one hop</td>
</tr>
<tr>
<td>(T_{ed})</td>
<td>Message encryption/decryption time on one hop</td>
</tr>
<tr>
<td>(T_a)</td>
<td>Authentication request service &amp; waiting time at the AP</td>
</tr>
<tr>
<td>(T_{sg})</td>
<td>Authentication request service &amp; waiting time at the proxy authentication server</td>
</tr>
<tr>
<td>(T_v)</td>
<td>Authentication request service and waiting time at the HAS</td>
</tr>
<tr>
<td>(T_{us})</td>
<td>Key encryption &amp; decryption time</td>
</tr>
<tr>
<td>(T_g)</td>
<td>Key generation time at the HAS</td>
</tr>
<tr>
<td>(T_{ts})</td>
<td>Transmission time for the session key to the other communication identities such as HA</td>
</tr>
<tr>
<td>(T_{rg})</td>
<td>Registration request service and waiting time at the HA</td>
</tr>
</tbody>
</table>

Here, \(\vec{x}_t\) is a vector defined as:

\[
\vec{x}_t^T = [T_{pr}, T_{tr}, T_{ed}, T_a, T_{sg}, T_v, T_{us}, T_g, T_{ts}, T_{rg}],
\]  

(3.15)

where all the time components are defined in Table 3.3. And, \(\vec{d}_{3,i}\) are the vectors defined as follows:

\[
\vec{d}_{1,1} = [2, 0, 1, 0, 0, 0, 0, 0, 0],
\]

\[
\vec{d}_{1,2} = [6, 2, 3, 0, 1, 0, 0, 0, 0],
\]

\[
\vec{d}_{1,3} = d_{1,4} = [8, 4, 4, 0, 2, 1, 0, 0, 0],
\]

\[
\vec{d}_{2,1} = [2, 0, 1, 0, 0, 0, 0, 0, 0],
\]

\[
\vec{d}_{2,2} = [2(N_h + 1), 2(N_h - 1), 3, 2(N_h - 2), 1, 0, 0, 0, 0],
\]

\[
\vec{d}_{2,3} = [2(N_h + 2), 2N_h, 4, 2(N_h - 2), 1, 1, 0, 0, 0],
\]

\[
\vec{d}_{2,4} = [2(N_h + 2), 2N_h, 4, 2(N_h - 2), 1, 2, 1, 1, 0],
\]

\[
\vec{d}_{3,1} = [2(N_h + 1), 0, 2, 2(N_h - 1), 0, 0, 0, 1],
\]

\[
\vec{d}_{3,2} = [2(N_h + 2), 2N_h, 3, 2(N_h - 2), 2, 0, 0, 0, 1],
\]

\[
\vec{d}_{3,3} = [2(N_h + 3), 2(N_h + 1), 4, 2(N_h - 2), 2, 1, 0, 0, 1],
\]

\[
\vec{d}_{3,4} = [2(N_h + 3), 2(N_h + 1), 4, 2(N_h - 2), 2, 2, 1, 1, 1].
\]
Call Dropping Probability

In Section 3.2.2, we consider a call is dropped during authentication if the waiting time for authentication is greater than a threshold value $T_{th}$, or an authentication failure happens. As defined in (3.8), we use a mean value from an experiment for the probability that authentication failure happens, i.e., $P_e$, due to the unknown distribution model of $P_e$. Therefore, to evaluate $P_\beta(i)$, ($\beta = 1, 2, 3$ and $i = 1, 2, 3, 4$), the authentication delay shown in (3.14) becomes the critical part.

In (3.14), we only consider the time variables, $T_{sq}$, $T_a$, $T_v$, and $T_{rg}$, as the random variables because the variance of the other time variables are small. $T_{ed}$ and $T_{us}$ are mainly related with the ability of computer and the message length, $T_{tr}$ is determined by the message length and the link speed, $T_{pr}$ is a function of the distance between two points, and $T_g$ is directly connected with the computer ability. In reality, the computer ability, message length, link speed, and distance between two points are all fixed. Therefore, we do not consider $T_{ed}$, $T_{tr}$, $T_{pr}$, $T_{us}$, and $T_g$ random variables. However, $T_a$, $T_{sq}$, $T_v$, and $T_{rg}$ are all related with the traffic load, queue length and service time, which are varied from time to time and have big variance.

Thus, to find $P_\beta(i)$ becomes to find the PDFs of the different combinations of $T_{sq}$, $T_a$, $T_v$, and $T_{rg}$ in $T_\beta(i)$. If we assume that: (1) $M/M/1$ queues are applied at APs, authentication servers, and HAs; (2) The PDFs of $T_{sq}$, $T_a$, $T_v$, and $T_{rg}$ are independent identical distribution (iid), then the PDF of $T_{sq}$, $T_a$, $T_v$, and $T_{rg}$, i.e., $w(t)$, can be shown as [26]:

$$w(t) = (\mu_s - \lambda_s)e^{-(\mu_s-\lambda_s)t}, \quad (3.17)$$

where $\mu_s$ and $\lambda_s$ are the service and arrival rates of authentication requests, respectively. Furthermore, the PDFs of the different combinations of $T_{sq}$, $T_a$, $T_v$, and $T_{rg}$ in $T_\beta(i)$, i.e., $f_{\beta,i}(t)$, can be expressed in (3.18), on next page as the components of a matrix $f(t)$.

In (3.18), $\beta$ and $i$ represent the row and column, respectively. $\Gamma(x) \triangleq \int_0^\infty s^{x-1}e^{-s}ds$, and $\xi = \mu_s - \lambda_s$. With these PDFs, $P_\beta(i)$ can be obtained in different cases.

To summarize, we have obtained authentication cost, delay, and call dropping probability for one authentication operation. However, in order to obtain the average authentication cost, delay, and call dropping probability defined in (3.6), (3.7), and (3.8), we still need to evaluate the arrival rates of different types of authentication requests,
\[ f(t) = \begin{bmatrix}
\xi e^{-\xi t} & \frac{\xi(\xi t)^3 e^{-\xi t}}{\Gamma(4)} & \frac{\xi(\xi t)^5 e^{-\xi t}}{\Gamma(6)} & \frac{\xi(\xi t)^5 e^{-\xi t}}{\Gamma(6)} \\
\xi e^{-\xi t} & \frac{\xi(\xi t)^3 e^{-\xi t}}{\Gamma(4)} & \frac{\xi(\xi t)^5 e^{-\xi t}}{\Gamma(6)} & \frac{\xi(\xi t)^5 e^{-\xi t}}{\Gamma(6)} \\
\frac{\xi(\xi t)^2 N_{h1} e^{-\xi t}}{\Gamma(2N_{h1}+1)} & \frac{\xi(\xi t)^2 N_{h2} e^{-\xi t}}{\Gamma(2N_{h2}+1)} & \frac{\xi(\xi t)^2 N_{h3} e^{-\xi t}}{\Gamma(2N_{h3}+1)} & \frac{\xi(\xi t)^2 N_{h4} e^{-\xi t}}{\Gamma(2N_{h4}+1)} \\
\frac{\xi(\xi t)^2 N_{h1} e^{-\xi t}}{\Gamma(2N_{h1}+2)} & \frac{\xi(\xi t)^2 N_{h2} e^{-\xi t}}{\Gamma(2N_{h2}+2)} & \frac{\xi(\xi t)^2 N_{h3} e^{-\xi t}}{\Gamma(2N_{h3}+2)} & \frac{\xi(\xi t)^2 N_{h4} e^{-\xi t}}{\Gamma(2N_{h4}+2)}
\end{bmatrix} \] (3.18)

that is, \( \lambda_\beta, \ (\beta = 1, 2, 3) \).

### 3.2.4 Arrival Rates of Authentication Requests

In our analysis, authentication requests are categorized into three types: intra-domain handoff authentication, session authentication, and inter-domain handoff authentication. Thus, we analyze the arrival rates of different types of authentication requests, i.e., \( \lambda_\beta, \ (\beta = 1, 2, 3) \), based on the mobility and traffic patterns of the MNs.

**Arrival Rate of Intra-Domain Handoff Authentication, \( \lambda_1 \)**

The intra-domain handoff authentication requests happen whenever an MN crosses the boundaries of subnets inside a network domain with an on-going service. In order to calculate the arrival rate of intra-domain handoff authentication requests, we categorize the calls into four types that happen in four events:

- \( Y_1 \) is the event that an MN starts a connection before entering the network domain, enters the network domain with the on-going connection and this connection ends before the MN moves out of the network domain.
- \( Y_2 \) is the event that an MN starts a connection within current network domain and this connection ends before the MN moves out of the network domain.
- \( Y_3 \) is the event that an MN starts a connection within current network domain and this connection ends after the MN moves out of the network domain.
- \( Y_4 \) is the event that an MN starts a connection before entering the network domain, enters the network domain with the on-going connection, and the connection ends after moving out of the network domain.
Then, the arrival rate of intra-domain handoff authentication requests, $\lambda_1$, can be written as:

$$
\lambda_1 = \lambda_u P_{r1}(\lceil N_{a1} \rceil - 1) + \lambda_u P_{r2}(\lceil N_{a2} \rceil - 1) + \lambda_u P_{r3}(\lceil N_{a3} \rceil - 1) + \lambda_u P_{r4}(\lceil N_{a4} \rceil - 1),
$$

(3.19)

where $P_{rj}$, $j = 1, 2, 3, 4$, is the probability that event $Y_j$ happens, $N_{aj}$, $j = 1, 2, 3, 4$, is the average number of subnets passed by an MN in current network domain in event $Y_j$.

The time diagrams of events, $Y_j$, $j = 1, 2, 3, 4$, are shown in Fig. 3.6. Therefore, $P_{rj}$, $j = 1, 2, 3, 4$, can be derived as follows.

According to the time diagram in Fig. 3.6.A, and denote $\Delta t = t^0_n - t^0_c$, we have:

$$
P_{r1} = \int_0^\infty P_r[I(t^0_c + \Delta t, t^0_c) = 1] \cdot P_r(T_D > \Delta t)d(\Delta t)
$$

(3.20)

$$
\cdot P_r(T_{Dr} \leq T_M),
$$

where $I(t^0_c + \Delta t, t^0_c)$ is the number of calls that arrive in time interval $[t^0_c, t^0_c + \Delta t]$. 

Figure 3.6: Time Diagrams of Events.
Second, $P_{r2}$ can be derived from Fig. 3.6.B as:

$$P_{r2} = P_r(T_D < T_{Mr}) \cdot P_r(t_{mr}^0 \leq t_c^0 < t_{mr}^0 + T_{Mr})$$

(3.21)

$$= \int_0^{\infty} f_{X_2}(t) \cdot \int_0^{\infty} \lambda_u t e^{-\lambda_u t} f_{Mr}(t) dt,$$

where $X_2 \triangleq T_{Mr} - T_D$, $f_{X_2}(t)$ and $f_{Mr}(t)$ are the PDFs of $X_2$ and $T_{Mr}$, respectively.

Moreover, we can obtain $P_{r3}$ from Fig. 3.6.C:

$$P_{r3} = P_r(T_D > T_{Mr}) \cdot P_r(t_{mr}^0 \leq t_c^0 < t_{mr}^0 + T_{Mr})$$

(3.22)

$$= \int_0^{\infty} f_{X_3}(t) \cdot \int_0^{\infty} \lambda_u t e^{-\lambda_u t} f_{Mr}(t) dt,$$

where $X_3 \triangleq T_D - T_{Mr}$, $f_{Mr}(t)$ is the PDF of $T_{Mr}$, $f_{X_3}(t)$ is the PDF of $X_3$.

Similar with $P_{r1}$, $P_{r4}$ can be determined from Fig. 3.6.D as follows:

$$P_{r4} = \int_0^{\infty} P_r[I(t_c^0 + \Delta t, t_c^0) = 1] \cdot P_r(T_D > \Delta t) d(\Delta t)$$

(3.23)

$$\cdot P_r(T_{Dr} > T_{M})$$

Figure 3.7: Time Diagram for Number of Subnets Passed by in One Call.

$T_D$: One Call Duration Time  
$t_i$: Residence Time in a Subnet $i$  
$T_{Mr}$: Residual Time of the Residence Time in a Network Domain  
$\triangledown$: Enter a Network Domain  
$\triangleright$: Leave a Network Domain  
$\blacklozenge$: Start a Call  
$\blacklozenge$: End a Call
After we obtain $P_{rj}$, $(j = 1, 2, 3, 4)$, in order to evaluate $\lambda_1$, we need to evaluate the average number of subnets passed by an MN in a network domain during one call in the events $Y_j$, i.e., $N_{aj}$, $(j = 1, 2, 3, 4)$, respectively. The time diagrams to evaluate $N_{aj}$ are shown in Fig. 3.7.

In events $Y_1$ and $Y_2$, the call duration time in the network domain are $T_{Dr}$ and $T_D$, respectively, which are exponential distribution, one special case of Gamma distributions. Therefore, $N_{a1}$ and $N_{a2}$ can be obtained as [23]:

$$N_{a1} = N_{a2} = \frac{\mu_r}{\eta},$$  \hspace{2cm} (3.24)

where $1/\eta$ is the average call duration time of the MN and $\mu_r$ is the average residence time of the MN in a subnet.

On the other hand, note that the call duration time in events $Y_3$ and $Y_4$, i.e., $T_{Mr}$ and $T_M$ are not Gamma distributions, thus we cannot obtain $N_{a3}$ and $N_{a4}$ with [23]. Therefore, we need to derive $N_{a3}$ and $N_{a4}$ next.

Fig. 3.7.C illustrates the time diagram that event $Y_3$ happens. From Fig. 3.7.C, the relationship between different time components can be written as follows:

$$T_{Mr} = t_r + \sum_{i=2}^{N_{a3}} t_i,$$  \hspace{2cm} (3.25)

where $T_{Mr}$ is the residual time of the residence time of an MN in a network domain.

Based on the relationship in (3.25), we can obtain:

$$F_{Mr}(s) = F_{t_r}(s)G_{N_{a3}-1}(z)|_{z=F_{t_r}(s)},$$  \hspace{2cm} (3.26)

Then, $N_{a3}$ can be obtained by:

$$N_{a3} = \frac{\partial G_{N_{a3}-1}(z)}{\partial z}|_{z=1} + 1$$

$$= \frac{2M^2-M-1}{12T_{Mr}{\mu_r}} + \frac{(M+1)(\gamma+1)}{\gamma} + 1.$$  \hspace{2cm} (3.27)

According to Fig.3.7.D, $N_{a4}$ is equal to the average number of subnets that an MN passes when the MN is roaming inside the network domain. Therefore, we have:

$$N_{a4} = N_{sn} = \sum_{j=1}^{M} j = \frac{M + 1}{2}.$$  \hspace{2cm} (3.28)
After we obtain $N_{a_4}$, we have obtained the average number of subnets passed by an MN in a network domain at event $Y_j$, $j = 1, 2, 3, 4$. Since we get the probabilities that event $Y_j$ occurs in previous part of this subsection, we can evaluate $\lambda_1$ by substituting the values of $P_{r_j}$ and $N_{a_j}$, $j = 1, 2, 3, 4$, into (3.19). Furthermore, in order to obtain $C(i)$, $T(i)$, and $P(i)$ defined in (3.6), (3.7), (3.8), we need to evaluate $\lambda_2$ and $\lambda_3$ next.

**Arrival Rate of Session Authentication, $\lambda_2$**

After an MN has moved into a network domain, a session authentication is initiated whenever a call arrives. Therefore, the arrival rate of session authentication requests for one MN, e.g. $\lambda_2$, is equal to the call arrival rate of an MN,

$$\lambda_2 = \lambda_u,$$  

(3.29)

where $\lambda_u$ is assumed to be the call arrival rate in (3.4).

**Arrival Rate of Inter-Domain Handoff Authentication, $\lambda_3$**

The inter-domain handoff authentication requests happen when an MN enters the network domain with an on-going service. Therefore, the arrival rate of inter-domain handoff authentication requests, $\lambda_3$, can be obtained by:

$$\lambda_3 = \lambda_u(P_{r_1} + P_{r_4}).$$  

(3.30)

Thus, we have obtained the arrival rates of authentication requests in the cases of intra-domain handoff authentication, session authentication, and inter-domain handoff authentication. Since two key aspects, i.e., the relationship between the security and system performance, and the relationship between the QoS metrics and traffic load, have been evaluated, the impact of authentication on security and the system performance can be observed clearly through $C(i)$, $T(i)$, and $P(i)$ in (3.6)~(3.8).

### 3.3 Numerical Results

In this section, we evaluate the effects of mobility and traffic patterns on authentication cost, $C(i)$, delay, $T(i)$, and call dropping probability, $P(i)$, at different security levels.
3.3.1 Assumptions and Parameters

The numerical results are presented based on the assumptions introduced in Sections 3.2.1 and 3.2.3. Of the assumptions in Section 3.2.1, we consider an MU roaming within a foreign network shown in Fig. 3.2. The mobility pattern of the MU is represented with the residence time in a subnet of the network domain, which is assumed to be Gamma distribution with the mean value $1/\mu_r$. The traffic patterns of an MU are represented by all arrival rate and call duration time. The call arrival rate is assumed to be Poisson process with mean value $1/\lambda_u$, and the call duration time is assumed to be exponential distribution with mean value $1/\eta$.

In Section 3.2.3, we further assume that $M/M/1$ queues are used at APs, authentication servers such as LAS and HAS, and HAs with service rate $\mu_s$ and arrival rate of authentication requests $\lambda_s$. Let $\xi = \mu_s - \lambda_s$. According to (3.17), the service and waiting time at an AP, authentication server, and HA, e.g., $T_a$, $T_{sq}$, and $T_v$, become random variables with identical exponential distribution with mean value of $1/\xi$. The parameters to evaluate the authentication cost and delay are shown in Table 4.4.

<table>
<thead>
<tr>
<th>Parameters for Evaluation on QoS Metrics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameters for Authentication Cost</td>
</tr>
<tr>
<td>$c_s$</td>
</tr>
<tr>
<td>10</td>
</tr>
<tr>
<td>Parameters for Authentication Delay</td>
</tr>
<tr>
<td>$T_{th}$</td>
</tr>
<tr>
<td>3s</td>
</tr>
<tr>
<td>Parameters for Random Variables</td>
</tr>
<tr>
<td>$\lambda_u$</td>
</tr>
<tr>
<td>0.1 $min^{-1}$</td>
</tr>
</tbody>
</table>

There are many ways to determine the values for the authentication costs. For example, the authentication cost for signaling can be measured with the number of messages, and the authentication cost for encryption can be measured with the number of CPU cycles. However, the most important problem here is how to make them consistent, i.e., the values of the costs can be compared with each other in the same scale. To solve this problem, we assume that the encryption/decryption cost on one hop, $c_p$, and the key generation cost, $c_g$, are normalized to a cost unit because they are the lightest
load compared to other costs and they have the similar operation in cryptography techniques [13, 29]. The values of other costs are determined by comparing to $c_p$ and $c_g$ with the time to finish the operation, i.e., we use the ratio of processing time to represent the authentication cost instead of the actual processing time. The reason is that the time needed to finish an operation represents the load of the server to complete it. However, we do not use the processing time to represent the cost directly because we do not want to confuse the authentication cost with the authentication delay and the authentication cost can be evaluated with many other ways.

When the maximum authentication message size is 4096 bytes [14], the transmission delay is about 20 milliseconds with the assumption of 2 Mbps link capacity [29]. The values of $T_{ed}$ and $T_g$ are obtained from existing research [29, 60]. By assuming one network domain is about 100 km$^2$ with radius 6 km, the value of the propagation time, $T_{pr}$, can be determined by the distance between two LASs as shown in Table 4.4.

### 3.3.2 Effects of Mobility Pattern at Different Security Levels

The effects of mobility pattern on the authentication cost, delay, and call dropping probability are shown in Figs. 3.8, 3.9, and 3.10. In these figures, we illustrate the relationships between the residence time of an MU in a subnet, authentication cost, delay, and call dropping probability, respectively.

In Fig. 3.8, authentication costs at different security levels decrease with the increase of the residence time of an MU in a subnet because the longer an MU stays in the subnets, the less the intra-domain handoff authentication requests. And, if the residence time of an MU approaches to infinity, the authentication cost will be stable on the session authentication cost because only session authentication exists in this case. Moreover, we can see that the security levels have different effects on the cost at the same residence time in a subnet. The higher the security level, the more the authentication cost because higher security levels impose more operations to provide secure services. For example, if we degrade the security level from 4 to 3, the authentication cost can be reduced up to 32%.

Fig. 3.9 reveals the effect of residence time on the authentication delay. As we can see, authentication delay decreases with the increase of the residence time of an MU in a subnet. Similar with the authentication cost, this trend is due to the decrease in
the intra-domain handoff authentication requests. And, the higher security levels cause more authentication delay because of more operations needed for more secure services. The improvement of authentication delay by changing security levels from 4 to 3 is around 0.1 seconds, which is around 18.2% of the authentication delay at security level 3 when the residence time of an MU in a subnet is 27 minutes.

The effect of call dropping probability in authentication is shown in Fig. 3.10. The call dropping probability increases with the increase of the residence time of an MU in a subnet. When the residence time of an MU in a subnet increases, the arrival rate of intra-domain handoff authentication requests will decrease. Then, the session authentication requests become the major part of authentication requests. Note that the call dropping probability for session authentication is much higher than that in intra-domain handoff authentication due to the longer authentication delay caused by remote authentication. The call dropping probability will approximate that in session authentication if the residence time of an MU approaches infinity. In other words, the upper bound of the call dropping probability can be achieved when authentication requests are all session authentication requests. Similar with the cost and delay, call dropping probability is greatly affected by the security levels. When the security level is leveraged from 3 to 4, call dropping probability increases about 0.45%, which is about 50% more than the call dropping probability at security level 3 when the residence time of an MU is 27 minutes.

**Effect of Traffic Load at Different Security Levels**

The effects of traffic pattern on the authentication cost, delay, and call dropping probability at different security levels are demonstrated in Figs. 3.11, 3.12, and 3.13.

Figs. 3.11 and 3.12 show that the authentication cost and delay increase with the call arrival rate of an MU. As shown in (3.6) and (3.7), the authentication cost and delay are proportional to the call arrival rate $\lambda_u$ since variables $\lambda_\beta$, $(\beta = 1, 2, 3)$ are proportional to $\lambda_u$. Moreover, a higher security level causes more cost and delay than a lower one. For example, if the security level is changed from 1 to 2, the authentication will be about 740% more cost and 29% more time than those at security level 1.

As for the call dropping probability at different call arrival rates, the call arrival rate of an MU does not affect the call dropping probability. As we can see
Figure 3.8: Authentication Cost vs. Residence Time in a Subnet.

Figure 3.9: Authentication Time vs. Residence Time in a Subnet.
in (3.8), \( P(i) \) for security level \( i \) is average call dropping probability computed in the cases of intra-domain handoff authentication, session authentication, and inter-domain handoff authentication. As shown in (3.19), (3.29, and (3.30, \( \lambda_\beta, (\beta = 1, 2, 3) \) are all proportional to \( \lambda_u \). Thus, \( \lambda_u \) disappears in (3.8), which is \( P(i) \)'s definition equation. Then, once the PDF of the call duration time and the mobility patterns of the MU are known, i.e., \( \eta, \mu_r, \) and \( \gamma \) are fixed, the call dropping probability of the MU is a constant at different call arrival rates shown in Fig. 3.13. However, the call dropping probability is different at different security levels. As we can see in Fig. 3.13, the call dropping probability at security level 4 is about 56% more than that at security level 3.

### 3.4 Conclusions

In this chapter, we investigated the impact of authentication on security and quality of service (QoS) in combination of mobility and traffic patterns, which is critical to deliver secure and efficient services in wireless networks such as wireless local area network (WLAN). We analyze the authentication cost, delay, and call dropping
Figure 3.11: Authentication Cost vs. Call Arrival Rate.

Figure 3.12: Authentication Time vs. Call Arrival Rate.
probability at different security levels in wireless networks based on a system model with a challenge/response mechanism. In the analysis, the mobility and traffic patterns are taken into account for the QoS. Therefore, this work provides a quantitative connection between the security and system performance with the concern of adaptation to various mobile environments, which further proposes a solid ground for an in-depth understanding of authentication impact, and demonstrates a framework for the future design of efficient authentication scheme in wireless networks.

Furthermore, since the analysis results show that the authentication efficiency is affected by the authentication distance, we propose enhanced authentication protocols for single-hop wireless networks next.
Chapter 4

Design of Authentication

Mechanisms in Single-Hop Wireless Networks

In this chapter, we propose enhanced authentication protocols for hierarchical authentication architecture (HAA) first. Corresponding enhanced authentication protocols are proposed for net-to-net and mobile-to-net authentication, respectively. Since the HAA is not well adapted to the network manageability and scalability, a new distributed authentication architecture as well as related authentication control schemes is proposed at the second Section.

4.1 Enhanced Authentication Protocols for Hierarchical Authentication Architecture

In this section, we introduce the system model of hierarchical authentication architecture first. Then, three enhanced authentication protocols are proposed to im-
prove the authentication efficiency for hierarchical authentication architecture.

4.1.1 System Model and Metrics

In order to reduce the time to establish SAs in wireless networks, many authentication architectures are proposed, which are dedicated to encrypting, transferring and verifying the credentials for MNs with static SAs [5, 11, 24, 43]. These solutions can be categorized into two types. One is distributed architecture; the other is hierarchical architecture. Here, we introduce hierarchical architecture because it is deployed widely in wireless networks and proposed by IETF as a recommended standard for mobile IP networks [24] due to its high scalability and manageability.

A hierarchical authentication architecture is proposed in [24], where each network only has one LAS and a proxy authentication server (PAS) is introduced to manage a group of authentication servers (ASs) with static SAs. The PASs are also organized in groups and are controlled by a higher-level PAS. Fig. 4.1 is an example of the hierarchical authentication architecture. In this example, AS1, AS2, AS3 and AS4 are ASs in the corresponding networks 1, 2, 3 and 4. AS1 and AS2 are controlled by PAS1, and AS3 and AS4 are managed by PAS2. The highest authentication proxy, PAS3, trusts two lower-level authentication proxies, PAS1 and PAS2. This architecture needs fewer inter-domain SAs, and the appearance of an AS only needs to inform one of the PASs, which greatly reduces the number of SAs between networks.

As shown in Fig. 4.1, when an inter-domain roaming MN, e.g., MN_{4\rightarrow1}, is visiting a foreign network, a mobile-to-net and a net-to-net SAs are needed to authenticate the MN. For the net-to-net authentication, since the direct net-to-net SA on hierarchical architecture does not exist, the credentials for authentication are transmitted by hop-by-hop encryption, which increases the work load of network and potential attacks such as man-in-middle attack. In addition, the mobile-to-net SA does not exist between the inter-domain roaming MN and LAS if the communication is not going on. Thus, each time the MN needs service, the LAS needs to relay the credentials to the HAS of the MN for verification, which also increases the workload for authentication.

Therefore, in order to improve the authentication efficiency, we need to propose intelligent management schemes for net-to-net and mobile-to-net SAs. After these two types of SAs are established well, the standard authentication protocols, such as
challenge/response authentication, can work efficiently, which also increases the compatibility of proposed schemes for many standards.

4.1.2 Net-to-Net Authentication with Traffic-Based Security Association

In this section, we propose an SA control scheme between wireless networks on hierarchical architecture. First, we illustrate the process of the proposed scheme. Then, we evaluate the authentication cost with this scheme and provide an optimal condition to minimize authentication cost based on user density, mobility and traffic patterns [49, 52].

**Design of Control Scheme**

In the proposed scheme based on hierarchical architecture, we investigate the authentication between two wireless networks. The scheme demands the LAS in a network periodically detecting the number of users, mobility and traffic patterns from different networks. The mobility and traffic patterns in the proposed scheme are presented with residence time and individual arrival rate of authentication requests, respectively. If the total number of authentication requests of MNs from foreign network $i$ within time $T$, denoted as $L_i(T)$, is greater than a threshold value $L^*_iT$, our proposed scheme will enable the LAS in current network $w$ to set up a direct SA with network $i$ using a
four-way handshake protocol in TLS. $L_i(T)$ can be written as:

$$L_i(T) = \sum_{k_i=1}^{K_i(T)} a_{k_i}(T)t_{k_i}(T)$$

(4.1)

where $k_i$ is the index of inter-domain roaming MNs from network $i$ to the network we are investigating, $T$ is observation time, $K_i(T)$ is the number of inter-domain roaming MNs that come from network $i$ and pass or stay at network $w$ within time $T$, $a_{k_i}(T)$ is an individual arrival rate of authentication requests generated by MN $k_i$, and $t_{k_i}(T)$ is the residence time of MN $k_i$ observed in time $T$.

An example of the proposed scheme working on hierarchical architecture is shown in Fig. 4.4, in which the LAS in network $w$ performs the proposed scheme as follows:

1. In network $w$, LAS periodically detects $K_i(T)$, $a_{k_i}(T)$ and $t_{k_i}(T)$ of the users from network $i$, which represent the mutual interests between network $w$ and $i$.

2. LAS in network $w$ calculates the value of $L_i(T)$ for network $i$ and compares it to the threshold value $L_{iT}^*$.

3. If $L_i(T)$ is greater than the threshold value $L_{iT}^*$, LAS in network $w$ initiates a four-way handshake protocol to set up a direct SA with network $i$ serving for next
time period $T$.

4. If $L_i(T)$ is less than the threshold value $L^*_iT$, LAS in network $w$ sends all of the authentication requests of users from network $i$ to the PAS trusted by the LAS in network $w$.

From these steps, we can see that $L^*_iT$ is critical to the proposed scheme with great effect on the performance. Therefore, we analyze the authentication cost with and without our control scheme in the following paragraphs, and derive the optimal value of $L^*_iT$ based on the analysis.

**Optimal Value of $L^*_iT$**

In order to obtain the optimal value of $L^*_iT$, first we need to derive the cost function with and without proposed authentication control scheme.

Assume the authentication costs for one inter-domain roaming authentication request with and without direct SA between networks $w$ and $i$ are $c^{(s)}$ and $c^{(l)}$, respectively. And, if we define $C^{(s)}$ and $C^{(l)}$ as the total authentication cost function with and without the proposed scheme in time $T$, respectively, $C^{(l)}$ and $C^{(s)}$ can be written as:

$$C^{(l)} = L_i(T)c^{(l)}, \quad C^{(s)} = L_i(T)c^{(s)} + c^{(l)} + c^{(m)}(T)$$

(4.2)

where $c^{(l)}$ is initialization cost to set up a direct SA between two networks, and $c^{(m)}(T)$ is SA maintenance cost in time $T$.

In our proposed scheme, when $C^{(l)}$ is less than or equal to $C^{(s)}$, we will initiate our scheme to set up a direct SA for efficient authentication. Therefore, the optimal value of $L^*_iT$ can be derived from (4.2) as follows:

$$L^*_iT = \frac{c^{(l)} + c^{(m)}(T)}{c^{(l)} - c^{(s)}}.$$  

(4.3)

**Authentication Cost without Proposed Scheme**

An authentication process with challenge/response mechanism involves the encryption of challenge value, transmission of encrypted data, decryption of data and verification of data [66]. When hop-by-hop SA is used for authentication, the data must be encrypted and decrypted one time on each SA. Therefore, assuming encryption, decryption and transmission cost are the same on different hops, $c^{(l)}$ can be written as:

$$c^{(l)} = 2n(c_e + c_d + c_t) + c_v,$$

(4.4)
where \( c_e \) is the encryption cost on one hop, \( c_d \) is the decryption cost on one hop, \( c_t \) is the transmission cost on one hop, \( c_v \) is the verification cost on HAS, and \( n \) is the number of hops between the LAS in network \( w \) and HAS in network \( i \).

**Authentication Cost with Proposed Scheme**

In our proposed scheme, because encryption and decryption are performed twice on one SA, \( c^{(s)} \) becomes:

\[
  c^{(s)} = 2(c_e + c_d) + 2nc_t + c_v. \tag{4.5}
\]

Here, \( c_e, c_d, c_t, \) and \( c_v \) are the same as those in (4.22).

However, in our proposed scheme, when the LAS in network \( w \) initiates a four-way handshake protocol to set up a direct SA with network \( i \), additional costs are induced, which include the initialization cost and the maintenance cost, which is a function of time. The initialization cost is considered the cost to establish direct SA between two networks with a four-way handshake protocol in TLS. And it is also related to the number of hops between two networks due to the transmission of confidential data for the direct SA. Thus, \( c^{(I)} \) is shown as:

\[
  c^{(I)} = 2n(c_e + c_d + c_t) + 2c_c. \tag{4.6}
\]

where \( n, c_e, c_d, \) and \( c_t \) are described in (4.22), while \( c_c \) is the cost to compute and generate parameters for the direct SA.

\( c^{(m)}(T) \) in the proposed scheme is the potential cost to keep the direct SA to continue authenticating the inter-domain roaming MNs. First, the storage of additional SA consumes some local space. Then, with the time increasing, the additional SA might compromise the authentication architecture by increasing the complexity of networks in terms of the number of SAs [2]. Furthermore, an extended existence of an additional SA provides attackers more chances to intrude the network. Therefore, \( c^{(m)}(T) \) can be written as:

\[
  c^{(m)}(T) = c_1 + c_o(T) \tag{4.7}
\]

where \( c_1 \) is the storage cost and \( c_o(T) \) is the other potential cost within time \( T \), which is defined as:

\[
  c_o(T) = \int_0^T c_2 e^{\alpha x} dx, \tag{4.8}
\]

where \( c_2 \) is a risk or cost value assigned to a security level. Since the concept of security level is widely used in network, environment and power risk assessment [61, 91], we
assign two values for two security levels to \( c_2 \). When an MN is in its home network, in general, the security level is low, which means the user trusts surroundings, and the risk value is low. When an MN is beyond its home network, the security level is high, and the risk value is accordingly high because the MN may face more threats such as denial of service due to incompatibility in a foreign network. \( \alpha \) is a coefficient to specify the speed of risk increase with \( t \).

**Evaluation of \( L_i(T) \) with Mobility and Traffic Patterns**

To evaluate \( L_i(T) \), we assume \( a_k \) and \( t_k \) are independent identical distribution (IID) to all the inter-domain roaming users from network \( i \). Then, \( a_k \) and \( t_k \) become \( a_i \) and \( t_i \). Furthermore, we assume \( a_i \) is a Poisson distribution with mean \( \lambda_i \), \( t_i \) is a Gamma distribution with mean \( \tau_i \). Since \( K_i(T) \) is the total number of inter-domain roaming users, it can be expressed as:

\[
K_i(T) = S_w U_i(T),
\]

where \( U_i(T) \) is the density function of inter-domain roaming users from network \( i \), which is supposed to be Gaussian distribution with mean \( m_i \), and \( S_w \) is the area of network \( w \) we are studying. Given these parameters, the mean value of \( L_i(T) \) can be evaluated as follows:

\[
E\{L_i(T)\} = S_w E\{U_i(T)\} E\{a_i\} E\{t_i\} = S_w m_i \lambda_i \tau_i.
\]

The condition in (4.3) then becomes the condition as follows:

- **Set up direct SA for network** \( i \), if \( S_w m_i \lambda_i \tau_i \geq \frac{c^{(1)} + c^{(m)}(T)}{c^{(1)} - c^{(s)}} \).
- **Break direct SA with network** \( i \), if \( S_w m_i \lambda_i \tau_i < \frac{c^{(1)} + c^{(m)}(T)}{c^{(1)} - c^{(s)}} \).

### 4.1.3 Mobile-to-Net Authentication with Cost-Based Security Association

We propose a local authentication control scheme in this section. First, we present an overview of the control scheme. Then, two critical parts of the scheme are introduced. One is the establishment of local SA; the other is to determine the optimal lifetime for the local SA and minimize the authentication cost, which also decreases the latency [51].
Overview of Mobile-to-Net Authentication with Cost-Based SA

The framework of the proposed scheme is illustrated in Fig. 4.6. When an inter-domain authentication request from a visiting MN comes to the LAS, the LAS first checks if a local SA exists for the MN. If the local SA exists, the LAS authenticates the roaming MN with this SA. Otherwise, the LAS checks if the residence time of the MN will be greater than a threshold value. There are many methods to estimate the residence time of an MN [16]. In our approach, we assume that the estimation result of the residence time exists. Then, if the residence time of the MN is greater than a threshold value, the LAS will authenticate the MN through the AAA architecture and generate a local SA for it. Otherwise, the LAS simply authenticates the MN through the AAA architecture and does not generate a local SA for it.

If a local SA is generated, we assign the value of residence time of the MN to the life time of the SA. The sequential authentication requests that will arrive within the life time of the local SA can be processed efficiently with the local SA without transmitting the credentials to the HAS of the MN. We focus on the establishment of the local SA in a mobile network for the roaming MN, which is highlighted in Fig. 4.6. The establishment of a local SA involves with two problems. One is how to distribute the key securely and efficiently; the other is how to determine the threshold value of residence time, i.e., the lifetime of an local SA, which is used to trigger proposed scheme.

Authentication and Local Security Association Establishment Protocol
The signaling diagram of the protocol to authenticate a roaming MN and establish a local SA for sequential authentication requests is shown in Fig. 4.4. As we can see in this diagram, when a foreign MN is requesting services in the local network, an authentication request is sent out to the LAS. The LAS replies a challenge, i.e., a random value, to the MN. The MN encrypts the challenge value with an SA shared with the HAS. The result of the value is a response value that is returned to the LAS. Because the LAS has no SA shared with the MN, the LAS relays the response value to the HAS of the roaming MN through the AAA architecture. The HAS of the MN decrypts the response value and compares the result with the challenge value transferred by the LAS. If these two values are matched, the MN is authenticated. A key $K_{u1}$ is generated with the SA shared between the MN and its HAS as follows:

$$K_{u1} = \text{HMAC} - \text{MD5}(K_0, \{R_1\|ID_{MN}\}),$$  \hspace{1cm} (4.11)

where $K_0$ is the pre-shared key in the SA between the MN and its HAS, $R_1$ is a random value of at least 64 bits. $ID_{MN}$ is the MN’s identity. $\text{HMAC} - \text{MD5}$ is a hash function implemented with MD5. The symbol $\|$ means that the two values are linked together. Then, the message that includes the following data is sent to the LAS:

$$\{K_{u1}, ALGORITHM, F_0, F_1, \{R_1, ALGORITHM, F_0\}_K_0\}_K_i,$$  \hspace{1cm} (4.12)
where $K_{u1}$ is the key generated for the local SA shared between the MN and the LAS, \textit{ALGORITHM} is the description of the algorithm for the local SA selected by the HAS that will be used for local authentication, $F_i$ is a random number used to avoid replay attack between AAA servers $i$ and $i-1$ in the AAA chaining servers shown in Fig. 4.5, $F_0$ is a random number used to avoid replay attack between the MN and the LAS, $K_0$ is the pre-shared key in the SA between the MN and its HAS, $K_i$ is the pre-shared key in the SA between AAA servers $i$ and $i-1$ in the AAA chaining servers shown in Fig. 4.5, the subscripts $K_0$ and $K_i$ mean that the data in the parenthesis are encrypted with $K_0$ or $K_i$, respectively.

![Diagram of Chaining AAA Servers](image)

**Figure 4.5**: Demonstration of Chaining AAA Servers.

When the LAS receives an authentication approval with the message shown in (4.12), the LAS decrypts the message with the key and algorithm in the SA shared with the upstream PAS and replies value $F_{n-1}$ to the PAS to avoid replay attack, where $n$ is the total number of chaining AAA servers between the LAS and HAS. Then, the LAS sends a message $\{R_1, \text{ALGORITHM}, F_0\}_{K_0} || \{\text{LIFETIME}\}_{K_{u1}}$ to the visiting MN, where \textit{LIFETIME} is the life time of the local SA and it is equal to the residence time of the MN. When the visiting MN receives the message $\{R_1, \text{ALGORITHM}, F_0\}_{K_0} || \{\text{LIFETIME}\}_{K_{u1}}$, the MN decrypts the first part of the message to obtain the value $R_1$ and generates the key $K_{u1}$ with (5.29). With key $K_{u1}$, the value of \textit{LIFETIME} is obtained. Then, the MN replies a value of $F_0 - 1$ to avoid replay attack.

After the above operations are finished, a local SA can be established at the visiting MN and the LAS as follows:
where UID is the unique user identification, which indicates the user for whom the local SA is used. In the local SA at the LAS, UID is the identification of the MN. In the local SA at the MN, UID is the identification of the LAS. SPI (Security Parameter Index) is the identification number of the association, which is used to differentiate the SAs uniquely. ALGORITHM is a description on a specific algorithm that should be used with this local SA. DIRECTION specifies the association used for packets arriving or leaving, KEY provides the encoding and decoding key for the authentication, which is $K_1$ in our proposed protocol. LIFETIME is a time period to keep the SA, which is determined and transferred by the LAS.

From the protocol, we can see that the security to distribute the key $K_{u1}$ is guaranteed. First, the messages transmitted between the AAA servers are encrypted with a pair of SAs with nonce technique. Thus, information secrecy and data integrity are provided and replay attack can be defeated. Second, the transmission of key $K_{u1}$ to the visiting MN from the HAS is protected through a random value $R_1$ with an SA shared between the MN and its HAS, which avoids direct key distribution on the unprotected medium and guarantees secure transmission of $K_{u1}$ from the HAS to the MN.

The threshold value of the residence time that triggers proposed scheme is critical to the authentication efficiency. When the residence time of the MN is greater than the threshold value, the authentication requests come within the life time of the local SA can be processed efficiently since the residence time of the MN is equal to the life time of the local SA. When the residence time of the MN is less than the threshold value, no local SA is established for the visiting MN. All the authentication requests sent by the visiting MN will be authenticated remotely through the AAA architecture, which will impose more burden to the network. Therefore, we propose an authentication cost function next to derive the threshold value of the residence time.

Threshold Value Based on Residence Time

In order to determine the threshold value of the residence time to trigger the proposed scheme, we evaluate the authentication cost with a accumulated authentication cost function first, which is related with the traffic and mobility patterns as well as the distance between the LAS and the HAS.
Accumulative authentication cost, $C_T$, is defined as the sum of signaling and encryption/decryption cost for all authentications sent by an MN when it is in a network. $C_T$ depends on the residence time and arrival rate of authentication requests of the MN because they determine the number of authentications in a network. In addition, the distance between the LAS and the HAS in terms of hops has a direct effect on $C_T$. Therefore, $C_T$ can be written as:

$$
C_T = c_{ms} \lambda \tau + (n - 1)c_{ss} + c_o,
$$

(4.13)

where $\lambda$ is the arrival rate of the authentications of the MN, $c_{ms}$ is the authentication cost on the hop between the MN and LAS, $c_{ss}$ is the authentication cost on the hop between authentication servers, $n$ is the number of hops between the MN and its HAS. $c_o$ is the maintenance cost for a local SA at the LAS. $\tau$ is a mean value of the residence time of the MN in a network. The first part of $C_T$ is the authentication cost for the MN after the establishment of local SA. The second part of $C_T$ is the cost to establish a local SA for the MN.

Assume that the existing authentication scheme is DIAMETER. In order to trigger our proposed scheme to replace it, $C_T$ should be at least less than the authentication cost with remote authentication in DIAMETER. Let $C_p$ be the authentication cost with remote authentication, $C_p$ can be obtained as:

$$
C_p = [(n - 1)c_{ss} + c_{ms}] \lambda \tau.
$$

(4.14)

By comparing (4.16) and (4.14), the threshold value of the residence time, i.e., $\tau_{th}$, of an MN can be carried out as:

$$
\tau_{th} = \frac{1}{\lambda} \left[ \frac{c_o}{(n - 1)c_{ss}} + 1 \right].
$$

(4.15)

If the residence time of an MN, i.e., $\tau$, is greater than the threshold, $\tau_{th}$, our scheme is triggered to establish the local SA for efficient authentication and assign the value of $\tau$ to the life time of the SA. Otherwise, DIAMETER is used at the LAS in the AAA architecture.
4.1.4 Mobile-to-Net Authentication with Risk-Aware Security Association

We propose a cost and risk aware authentication protocol with local SA control in this section. First, we provide an overview of the control process. Then, we use the same protocol introduced in previous section to establish a local SA. Finally, we determine the optimal life time of the local SA to minimize the combination of authentication cost and risk [50].

Overview of Mobile-to-Net Authentication with Risk-Aware SA

The overview of our proposed protocol is illustrated in Fig. 4.6. When an inter-domain authentication request from a visiting MN comes to the LAS, the LAS first checks if a local SA exists for the MN. If the local SA exists, the LAS authenticates the roaming MN with this SA. Otherwise, the LAS relays the credentials of the roaming MN through AAA architecture to the HAS for authentication. When the authentication is granted, a local SA is generated for the roaming MN. The sequential authentication requests arriving within the life time of the local SA will be processed efficiently with the local SA.
Many papers provide the authentication protocols with shared SA between an MN and an LAS [66]. Therefore, we do not focus on the authentication of the MN locally with shared SA. Instead, we focus on the establishment of the local SA in a mobile network for the roaming MN, which is highlighted in Fig. 4.6. The establishment of a local SA involves with two problems. One is how to distribute the key securely and efficiently; the other is how to determine the life time of the local SA to minimize the authentication cost and risk. Since the protocol to distribute local SA has been introduced in previous section, we just focus on the latter issue.

Optimal Life Time

In order to determine the optimal life time for the local SA, we evaluate the total authentication cost with a cost function in combination of risk, which is related with the life time of the local SA.

The authentication cost is defined as the signaling cost for one authentication request sent by a visiting MN in a foreign network domain. The total authentication cost, \( C(T) \), is defined as the sum of the authentication cost to process all the authentication requests sent by a visiting MN in a foreign network domain. In \( C(T) \), we consider the risk that one SA is being cracked as part of the authentication cost because an additional SA, i.e., the local SA, increases the possibility that the security is compromised due to unpredicted events such as unknown attacks. Then, \( C(T) \) can be written as:

\[
C(T) = \begin{cases} 
\lambda \tau c_m & \text{if } T = 0 \\
\frac{\tau}{T} (\lambda T c_n + c_r e^{\beta T}) + \frac{\tau c_r}{T} + c_m & \text{if } 0 < T \leq \tau \\
\lambda \tau c_n + c_r e^{\beta T} & \text{if } T > \tau
\end{cases}
\]

(4.16)

where \( \lambda \) is the arrival rate of session authentication requests, which is defined as the authentication initiated to begin a new service for the MN. Therefore, \( \lambda \) is equal to the call arrival rate of the MN. \( T \) is the life time of the local SA. Once we determine the life time, we use the same value of \( T \) whenever we refresh the local SA. \( c_n \) is the authentication cost for one authentication with local SA, \( c_m \) is the authentication cost for one authentication with remote authentication to the HAS of the MN, \( c_r \) is the cost to compensate the risk that one SA is cracked. For example, if the crack of the local SA induces data loss, the compensation cost is the cost to recover the original data from the backup data. \( \beta \) is an factor of the increasing speed of the risk, \( \tau \) is the residence time of the MN in the network, and \( c_c \) is the signaling cost to refresh a local SA.
The first line of \( C(T) \) in (4.16) is the total authentication cost without the local SA. In this case, the life time of the local SA is set to 0. Therefore, when a session authentication request arrives, the LAS must authenticate the visiting MN from its HAS because of the lack of local credentials. The total authentication cost is equal to the sum of the cost for the authentication requests sent by the MN when it resides in current network domain.

The second line of \( C(T) \) in (4.16) is the total authentication cost with our proposed protocol if \( 0 < T \leq \tau \). \( \lambda T c_m + c_r e^{\beta T} \) is the total authentication cost and the risk that a local SA is being cracked within the life time \( T \). Once a local SA is established through our proposed protocol, the authentication requests arrive within the life time \( T \) of the SA can be processed locally. At the same time, the existence of the local SA has the risk of being hacked, which is increased with the existence time of the local SA. In our cost function, we use \( c_r e^{\beta T} \) to present this risk. To decrease the risk, we refresh the local SA when the life time of the SA expires. Therefore, if \( 0 < T \leq \tau \), the times to refresh an SA is \( \frac{T}{\tau} \) because the LAS refreshes the local SA every \( T \) minutes, thus the signaling cost to refresh the local SA is \( \frac{T}{\tau} c_c e^{\beta T} \). And the total authentication cost should include the signaling cost to establish the local SA for the first time, i.e., \( c_m \).

The third line of \( C(T) \) in (4.16) is the total authentication cost with the proposed protocol if \( T > \tau \). In this case, it is clear that the total authentication cost is equal to the sum of authentication cost for all the authentication requests sent by the MN in the foreign network domain and the cost to compensate the risk that the SA is being cracked in time \( T \).

The authentication cost for one local authentication request, \( c_n \), can be evaluated with the number of signalings. As shown in Fig. 4.4, \( c_n = 4 \). Similarly, \( c_m \) can be represented with the number of signalings between the visiting MN and its HAS. Therefore, \( c_m = 4 + 2 \times n \), where \( n \) is the number of hops between the LAS and the HAS. For \( c_c \), we evaluate it with the number of signalings to refresh a local SA. When the life time expires, the local SA can be refreshed with two signalings. One is sent by the LAS to notify the MN with necessary new data such as new key; the other is sent by the MN to confirm the reception of the message. So \( c_c = 2 \). For the cost to compensate the risk that a local SA is being cracked, i.e., \( c_r \), we evaluate it with the number of destroyed records of the MNs caused by the crack of the local SA. We assume one
local SA only affects one record of the visiting MN. Therefore, $c_r = 1$. In our proposed protocol, we ask the MN to save its traffic and mobility patterns in its profile in terms of call arrival rate, $\lambda$, and average residence time of the MN in a subnet, $T_r$. When the MN needs authentication, these data should be sent to the LAS. Then, the call arrival rate, $\lambda$, and average residence time of the MN in a subnet, $\bar{T}_r$, can be obtained from the MN’s profile. For the residence time of the MN in a network domain, i.e., $\tau$, we use its average value, $\bar{\tau}$.

In our proposed protocol, we assign a life time to the local SA that meets the condition $0 < T \leq \tau$. Then, the optimal value $C^*(T^*)$ can be obtained by taking derivative of $C(T)$ with respect of $T$ as follows:

$$C^*(T^*) = \frac{\bar{\tau}}{T^*} (\lambda T^* c_n + c_r e^{\beta T^*}) + \frac{\tau c_r}{T^*} + c_m,$$

(4.17)

$T^*$ is the solution of $e^{\beta T^*} (\beta T^* - 1) = c_c / c_r$, which can be obtained with discrete method.

By calculating $T^*$ at the LAS and transmitting it to the visiting MN, the local SA can be established securely and efficiently, and the authentication cost with the consideration of risk evaluation, mobility and traffic patterns can be minimized, simultaneously.

4.1.5 Numerical Results

In this section, we show the numerical results for our proposed three authentication protocols, i.e., net-to-net authentication with traffic based SA, mobile-to-net authentication with cost based SA, and mobile-to-net authentication with risk-aware SA.

Net-to-Net Authentication with Traffic Based SA

In our simulation, we have twenty wireless networks communicating with the wireless network $w$, in which we evaluate the authentication cost with the proposed scheme. In the simulation architecture, $LAS_w$ is the LAC associated with the HASs in twenty wireless networks. A large number of MNs from different wireless networks are roaming inside the investigated network.

We generate the user mobility patterns, user density functions and user traffic patterns for different networks with the parameters shown in Table 4.5. The area of the network $w$ is assumed to be $100 km^2$, and the cost components discussed in our
Table 4.1: Simulation Parameters for Different Networks.

<table>
<thead>
<tr>
<th>netID(i)</th>
<th>hops</th>
<th>( m_i/(m^2) )</th>
<th>( \lambda_i ) (/user/minute)</th>
<th>( \tau_i ) (minutes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>0.3</td>
<td>0.2</td>
<td>30</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
<td>0.1</td>
<td>0.4</td>
<td>25</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>0.6</td>
<td>0.06</td>
<td>20</td>
</tr>
<tr>
<td>4</td>
<td>6</td>
<td>0.7</td>
<td>0.5</td>
<td>40</td>
</tr>
<tr>
<td>5</td>
<td>8</td>
<td>0.6</td>
<td>0.1</td>
<td>10</td>
</tr>
<tr>
<td>6</td>
<td>7</td>
<td>0.4</td>
<td>0.15</td>
<td>15</td>
</tr>
<tr>
<td>7</td>
<td>12</td>
<td>0.7</td>
<td>0.4</td>
<td>45</td>
</tr>
<tr>
<td>8</td>
<td>10</td>
<td>0.6</td>
<td>0.3</td>
<td>30</td>
</tr>
<tr>
<td>9</td>
<td>4</td>
<td>0.7</td>
<td>0.4</td>
<td>15</td>
</tr>
<tr>
<td>10</td>
<td>6</td>
<td>0.2</td>
<td>0.3</td>
<td>20</td>
</tr>
<tr>
<td>11</td>
<td>3</td>
<td>0.7</td>
<td>0.25</td>
<td>10</td>
</tr>
<tr>
<td>12</td>
<td>15</td>
<td>0.6</td>
<td>0.5</td>
<td>35</td>
</tr>
<tr>
<td>13</td>
<td>5</td>
<td>0.1</td>
<td>0.15</td>
<td>20</td>
</tr>
<tr>
<td>14</td>
<td>9</td>
<td>0.2</td>
<td>0.35</td>
<td>40</td>
</tr>
<tr>
<td>15</td>
<td>14</td>
<td>0.2</td>
<td>0.45</td>
<td>35</td>
</tr>
<tr>
<td>16</td>
<td>20</td>
<td>0.8</td>
<td>0.4</td>
<td>25</td>
</tr>
<tr>
<td>17</td>
<td>8</td>
<td>0.6</td>
<td>0.3</td>
<td>30</td>
</tr>
<tr>
<td>18</td>
<td>9</td>
<td>0.2</td>
<td>0.2</td>
<td>15</td>
</tr>
<tr>
<td>19</td>
<td>5</td>
<td>0.3</td>
<td>0.25</td>
<td>10</td>
</tr>
<tr>
<td>20</td>
<td>3</td>
<td>0.6</td>
<td>0.15</td>
<td>10</td>
</tr>
</tbody>
</table>

The simulation results are shown in Figures 4.7-4.10. In Fig. 4.7, the authentication cost with the control scheme is about 17% less than the cost without control.
scheme. In Fig. 4.8, we can see that although both of the authentication costs with and without the control scheme are increasing with arrival rate of inter-domain authentication requests, the cost with the control scheme is less than that without the control scheme. Fig. 4.9 shows the impact of the number of hops between networks on the authentication cost with and without our control scheme. We can see that the authentication cost with our control scheme is always less than that without our control scheme although increasing hops cause the authentication costs to increase. In Fig. 4.10, the authentication cost increases with the increase of residence time of the MNs in both cases. However, the cost with our proposed scheme is far less than the cost without the control scheme. The improvement in these figures comes from the establishment of direct SAs between networks, which reduces the multiple encryption and decryption cost during the authentication.

![Figure 4.7: Authentication Cost in Simulation.](image)

**Mobile-to-Net Authentication with Cost Based SA**

In our simulation, the parameters needed to evaluate the authentication cost and latency are shown in Table 4.4. We use the number of signaling messages for authentication to evaluate the authentication cost $c_{ms}$ and $c_{ss}$. Therefore, from Fig. 4.4,
Figure 4.8: Authentication Cost vs. Arrival Rate of Authentication Requests.

Figure 4.9: Authentication Cost vs. Increasing Number of Hops.
we can obtain $c_{ms} = 4$ and $c_{ss} = 2$. As for authentication latency, since the transmission delay is dominant in secret-key-based authentication, we use the delay to transmit an authentication message to represent the authentication latency. When the maximum authentication message size is 4096 bytes [14], the transmission delay for one signaling message is about 16 milliseconds with the assumption of 2 Mbps link capacity [29]. Therefore, $t_{ms} = 64 ms$ and $t_{ss} = 32 ms$. The default values of $n$, $\lambda$, and $\tau$ are set as 4, 0.3, and 10, respectively. In the following figures, when we change any one of these three parameters, the other two will be kept at the default values.

Based on these parameters, we evaluate and compare authentication latency per operation and accumulative authentication cost with DIAMETER. The results are shown in Figures 4.11, 4.12, and 4.13. In Fig. 4.11, the improvement of authentication latency

<table>
<thead>
<tr>
<th>$c_{ms}$</th>
<th>$c_{ss}$</th>
<th>$t_{ms}$ (ms)</th>
<th>$t_{ss}$ (ms)</th>
<th>$n$</th>
<th>$\lambda$ (min$^{-1}$)</th>
<th>$\tau$ (min)</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>2</td>
<td>64</td>
<td>32</td>
<td>4</td>
<td>0.3</td>
<td>10</td>
</tr>
</tbody>
</table>

Figure 4.10: Authentication Cost vs. Increasing Residence Time.

Table 4.3: Simulation Parameters for the Mobile-to-Net Authentication with Cost-Based SA.
Figure 4.11: Authentication Time vs. Residence Time of an MN.

Figure 4.12: Authentication Cost vs. Number of Hops.
latency per operation with proposed scheme is about 68%. The improvement comes from the use of local SA to authenticate the visiting MN. Since the remote authentication is changed to local authentication, the authentication latency is reduced greatly. In Fig. 4.12, we can see that the improvement of accumulative authentication cost is increasing with the increase of the number of hops between the MN and its HAS. At the point of nine hops, the proposed scheme costs 50% less than the DIAMETER. The benefit also comes from the utilization of local SA for authentication. In Fig. 4.13, although the authentication cost increases with the increase of arrival rate of authentication requests in both cases, the cost with proposed control scheme is less than that with DIAMETER with 40% improvement at the point of \( \lambda = 0.5 \).

**Mobile-to-Net Authentication with Risk-Aware SA**

The simulation scenario and parameters are set to the same as those in previous cost-based SA control scheme between MNs and ASs.

We assume that there are 100 subnets in a network domain where the visiting MU is roaming. The distance, \( n \), is represented in terms of hops between the LAS and the HAS, which is set to 10. The related authentication costs in terms of number of signalings or the number of records associated with the risk of one local SA are 24, 4,
Table 4.4: Simulation Parameters for the Mobile-to-Net Authentication with Risk-Aware SA.

<table>
<thead>
<tr>
<th>( M )</th>
<th>( n )</th>
<th>( c_m )</th>
<th>( c_n )</th>
<th>( c_r )</th>
<th>( c_c )</th>
<th>( T_r ) (minutes)</th>
<th>( \tau ) (minutes)</th>
<th>( \lambda ) (per minute)</th>
<th>( \beta )</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>10</td>
<td>24</td>
<td>4</td>
<td>1</td>
<td>2</td>
<td>10</td>
<td>505</td>
<td>0.3</td>
<td>0.8</td>
</tr>
</tbody>
</table>

1, and 2 for \( c_m \), \( c_n \), \( c_r \), and \( c_c \), respectively, which have been described in Section 4.1.4. By assuming that \( T_r = 10 \) minutes, \( \tau \) can be obtained as 505 minutes [83]. The call arrival rate of the visiting MU is assumed to be 0.3 times per minute. The coefficient \( \beta \) in our proposed protocol is assumed to be 0.8, which can be adjusted according to the knowledge of the risk in the environments. For example, if the historical data show that many attacks succeeded recently, the environment can be thought unsafe, and the value of \( \beta \) can be set to a big value, which indicates the risk increases very fast. If the historical data show that the attacks to the local SAs did not succeed frequently, the value of \( \beta \) can be set to a small value to demonstrate a slow increase of the risk with the time.

We evaluate the effects of residence time, call arrival rate, number of hops between LAS and HAS, and number of subnets in a network domain on the total authentication cost. The numerical results are shown in Fig. 4.14, Fig. 4.15, Fig. 4.16, and Fig. 4.17, respectively.

In Fig. 4.14, the total authentication cost is increasing with the increase of residence time of a visiting MN in a subnet. The longer the MN stays in current network domain, the more authentication requests the MN sends. Therefore, the total authentication cost increases due to the large amount of authentication requests. This increasing trend is same to DIAMETER and our proposed protocol. However, the total authentication cost with our proposed protocol outperforms that with DIAMETER because the authentication with local SA avoids the remote authentication signalings. The improvement is about 34.3% when \( T_r = 6 \) and 34.8% when \( T_r = 12 \).

Fig. 4.15 shows a trend in both DIAMETER and our proposed protocol that the total authentication cost increases with the increase of call arrival rate of a visiting MU. Whenever a call is initiated, an authentication request is sent out. Then, the number of authentication requests in current network domain increases with the call arrival rate. Accordingly, the total authentication cost increases with the increase of call arrival rate. In some cases that \( \lambda < 0.18 \), the total authentication cost with DIAMETER
Figure 4.14: Total Authentication Cost vs. Residence Time of an MN in a Subnet.

Figure 4.15: Total Authentication Cost vs. Call Arrival Rate of a Visiting MU.
is less than our proposed protocol. It is because the proposed protocol needs to establish, refresh, and keep a local SA, which takes costs. If the call arrival rate, i.e., the number of authentication requests during residence time in a network domain, is too small, the costs spent with the proposed protocol is not worthy. However, if the call arrival rate is bigger than 0.18 times per minute, the proposed protocol economizes much authentication cost. The improvement of authentication costs increases with the call arrival rate. When $\lambda = 0.3$, the improvement of our proposed protocol is about 34.7%.

![Figure 4.16: Total Authentication Cost vs. Distance between LAS and HAS.](image)

The relationship between the total authentication cost and the number of hops between LAS and HAS is shown in Fig 4.16. We can see that the authentication cost with DIAMETER increases with the increase of the number of hops between LAS and HAS, while the authentication cost with the proposed protocol remains constant with the increase of the number of hops between LAS and HAS. The reason is that whenever a session authentication is initiated, the challenge/response authentication in DIAMETER needs the LAS to authenticate the MU from the HAS, which requires remote delivery of the credentials. Therefore, the authentication cost with DIAMETER increases with the number of hops between the LAS and HAS. In the proposed protocol,
after the first authentication, the rest of the authentication requests for the visiting MU become local authentication, which has no relation with the number of hops between the LAS and the HAS. Therefore, the authentication cost with proposed protocol remains constant.

![Total Authentication Cost vs. Number of Subnets](image)

**Figure 4.17**: Total Authentication Cost vs. Number of Subnets in a Network.

In Fig. 4.17, we illustrate the effect of number of subnets in a network domain on the total authentication cost. The total authentication cost increases in DIAMETER and our proposed protocol with the increase of number of subnets in a network domain. When the number of subnets in a network domain increases, the residence time of an MN in a network domain increases if the other conditions such as residence time in a subnet do not change. Therefore, the number of authentication requests becomes big with the increase of residence time in a network domain. Accordingly, the total authentication cost increases. However, the total authentication cost with our proposed protocol is far less than that with DIAMETER because of the implementation of local authentication with the local SA. The improvement with our proposed is about 34.8% when \( M = 100 \).

In summary, we propose net-to-net and mobile-to-net authentication protocols
with various control scheme of SAs. The final purpose of these schemes is to improve the authentication efficiency with adaptation to the mobility, traffic and security of MNs. Comparing with the authentication without the control schemes, the proposed schemes demonstrate that the authentication cost is greatly reduced with the increase in the number of hops, arrival rate of authentication requests, and residence time of MNs.

Furthermore, considering that the hierarchical authentication architecture is not sufficient to the manageability and security of networks, we propose a dynamic authentication architecture and authentication control schemes next. The final purpose of the proposed architecture and schemes is to improve the authentication efficiency. Meanwhile, the manageability and security of wireless networks can be improved in terms of the number of SAs between networks.

### 4.2 Dynamic Authentication Architecture and Control Schemes

In previous section, we have improved the authentication efficiency greatly for HAA. Considering that the HAA is using static SA between networks for hop-by-hop authentication, the manageability and security are not well adapted to the traffic, mobility, and security requirements. Thus, we propose a dynamic authentication architecture in this section. Based on this architecture, two control schemes are necessary. One is a control scheme between LACs for net-to-net authentication; the other is a control scheme for mobile-to-net authentication [55].

#### 4.2.1 New Authentication Architecture

In the new authentication architecture, each network has only one licensed authentication center (LAC), which is an authority responsible for authenticating the MNs to be served in a wireless network. An LAC shares static SAs with access routers in an autonomous network. Here, an access point (AP) is a function unit for transmitting data. All the MNs subscribing to the service in a wireless network are trusted by a home authentication server (HAS) in their home network. An HAS is an authentication center that only takes charge of authentication for the MNs that subscribe services in its network. In our design, the LAC and HAS are combined together. All the
distributed LACs are connected to each other with dynamic SAs, which are introduced in Section 2.1.3. All MNs that subscribe the service in a network share static SAs with the LAC in their home network. And, all the visiting MNs will share dynamic SAs with the LAC in foreign networks.

An example of the proposed authentication architecture is shown in Fig. 4.18. In this example, there are three wireless networks $A$, $B$ and $C$. $LAC_A$, $LAC_B$, and $LAC_C$ are LACs and HASs in networks $A$, $B$, and $C$, respectively. $LAC_A$, $LAC_B$, and $LAC_C$ are associated with each other through dynamic SAs. $A_1$ and $A_2$ are two APs in network $A$, sharing static SAs with $LAC_A$. $B_1$ and $B_2$ are also two APs in network $B$. Static SAs are maintained between these two APs and $LAC_B$. In the third network $C$, APs, $C_1$ and $C_2$, are trusted by $LAC_C$. All MNs that subscribe the service in network $A$ share static SAs with $LAC_A$ and dynamic SAs with $LAC_B$ and $LAC_C$. The similar trust relationships exist for the MNs that subscribe the service in networks $B$ and $C$.

Since the SAs between LACs/networks are dynamic in this architecture, the number of SAs is of great interest, which will demonstrate the manageability of networks. Therefore, we calculate the number of SAs between networks next, and compare it to the number of SAs between networks in distributed and hierarchical architectures. Let $N_N$ be the number of SAs between LACs in the proposed architecture, it can be written...
as:

\[ N_N = \lim_{T_p \to \infty} \frac{\sum_{i=0}^{M_N-1} \sum_{j=0, j \neq i}^{M_N-1} \int_0^{T_p} n_{ij}(t) dt}{T_p}, \]  

(4.18)

where \( M_N \) is the number of LACs in our architecture, \( T_p \) is an observation time within which we count the number of SAs, \( i \) and \( j \) are the indices of the LACs within the wireless networks, \( n_{ij}(t) \) is the actual number of SAs from the LAC in network \( i \) to the LAC in network \( j \). Then, let \( t_{\text{max}} \) be the maximal lifetime of dynamic SA between any two wireless networks, \( A_{\text{max}} \) be the maximal arrival rate of inter-domain authentication requests, and \( t_{\text{max}}A_{\text{max}} < 1 \), we can prove that \( N_N = t_{\text{max}}A_{\text{max}}M_N(M_N - 1) \).

**Proof:** Assume \( t_{ij} \) is the lifetime of the dynamic SA from the LAC in networks \( i \) to the LAC in network \( j \), \( A_{ij} \) is the arrival rate of authentication requests from the LAC in networks \( i \) to the LAC in network \( j \), then according to the definitions of \( t_{\text{max}} \) and \( A_{\text{max}} \), we have:

\[ t_{\text{max}} = \max_{0 \leq i \leq M_N - 1, 0 \leq j \leq M_N - 1} \{t_{ij}\}, \text{ and } A_{\text{max}} = \max_{0 \leq i \leq M_N - 1, 0 \leq j \leq M_N - 1} \{A_{ij}\}. \]  

(4.19)

\[ n_{ij}(t) \]

\[ t \]

\[ t_{ij} \]

\[ 0 \]

\[ 1 \]

\[ t_{ij}A_{ij} \]

\[ 1/A_{ij} \]

\[ t_{\text{max}}A_{\text{max}} < 1, \]  

Figure 4.19: Number of SAs from LAC \( i \) to LAC \( j \).

Because \( t_{\text{max}}A_{\text{max}} < 1 \), we have \( t_{ij}A_{ij} < 1 \) and \( t_{ij} < 1/A_{ij} \) for any \( i \in [0, M_N - 1], j \in [0, M_N - 1] \) when \( i \neq j \), which means next authentication request has no effect on the current dynamic SA. In this case, when an authentication request arrives at LAC \( i \), it requires a dynamic SA from LAC \( i \) to LAC \( j \) for authentication. After the authentication, the dynamic SA is broken down. Therefore, during time \( t_{ij} \), the value of \( n_{ij}(t) \) is 1 for there only exists one dynamic SA from the LAC \( i \) to the LAC \( j \), and the value of \( n_{ij}(t) \) is 0, if the dynamic SA does not exist. Thus, the variation of \( n_{ij}(t) \) with respect to time \( t \) is like the curve in Fig. 4.19. Then, the number of SAs becomes:
\[ N_N = \lim_{T_p \to \infty} \frac{\sum_{i=0}^{M_N-1} \sum_{j=0, j \neq i}^{M_N-1} \int_0^{T_p} n_{ij}(t) \, dt}{T_p} \]
\[ = \sum_{i=0}^{M_N-1} \sum_{j=0, j \neq i}^{M_N-1} t_{ij} A_{ij} \]
\[ \leq \sum_{i=0}^{M_N-1} \sum_{j=0, j \neq i}^{M_N-1} t_{max} A_{max} = t_{max} A_{max} M_N (M_N - 1). \]  

(4.20)

Note that, in reality, the unit of authentication time is milliseconds between two servers [28, 29], and the arrival rate of inter-domain authentication requests in one second is small [29]. Therefore, most of the time, the condition \( t_{max} A_{max} < 1 \) can be satisfied. The average number of SAs in our architecture is far less than that needed in the distributed architecture introduced in Chapter 2.2.1. As for the hierarchical architecture, let \( M_H = M_N \), we have:

\[ \frac{N_N}{N_H} = \frac{t_{max} A_{max} M_N (M_N - 1)}{2^{v-1}(M_H - 1)} = \frac{t_{max} A_{max} (v - 1)}{2^{v}} M_N. \]  

(4.21)

If \( \frac{t_{max} A_{max} (v - 1)}{2^{v}} M_N < 1 \), the number of SAs in our proposed architecture is less than that in hierarchical architecture. The improvement in the number of SAs is \( \frac{t_{max} A_{max} (v - 1)}{2^{v}} M_N \). Since the number of SAs is related to the management effort of networks [2], the proposed architecture provides benefits of great manageability with reduced number of SAs.

Note that we need to manage two types of dynamic SA: one is between LACs; the other is between an LAC and visiting MNs. Therefore, we develop a scheme to manage the dynamic SAs between networks in Section 4.2.2 and a control scheme between an LAC and MNs in Section 4.2.3 to improve the composite performance of system with the concerns of bandwidth efficiency, risk assessment and manageability of networks in terms of the number of SAs.

4.2.2 Net-to-Net Authentication with Traffic-Based Security Association

Since the SA control scheme for efficient authentication between wireless networks has been developed before based on hierarchical architecture, we implement a
scheme with minor difference from the scheme on the evaluation of authentication cost, which is evaluated at the proposed architecture. By substituting the corresponding authentication cost into the SA control scheme between networks on hierarchical architecture, the SA control scheme between networks on proposed new architecture is able to achieve the same improvement in terms of minimizing the authentication cost.

Thus, on the proposed new architecture, we need to evaluate the variables, $c(s)$, $c(I)$ and $c(m)(T_{ob})$.

**Authentication Cost Per Request with a Direct SA between LACs, $c(s)$**

An authentication process with challenge/response mechanism [66] involves with the encryption of challenge value, transmission of encrypted data, decryption of data and verification of the data. Therefore, $c(s)$, can be written as follows:

$$c(s) = 2(c_e + c_d + c_t) + c_v,$$ (4.22)

where $c_e$ is the encryption cost of one transmission between LACs, $c_d$ is the decryption cost of one transmission between LACs, $c_t$ is the transmission cost of one transmission between LACs, $c_v$ is the verification cost on HAS.

**Initialization and Maintenance Costs to Establish a Dynamic SA between LACs, $c(I)$**

In our proposed scheme, when an LAC in network $w$ decides to initiate four-way handshake protocol to set up a direct SA with network $i$, additional costs will be induced, which include initialization cost and maintenance cost. The initialization cost in our proposed scheme is considered the cost to establish direct SA between two networks with four-way handshake protocol in TLS. And it is also related with the number of hops between two networks due to the transmission of confidential data for the direct SA. Define the initialization cost as $c(I)$, it can be shown as:

$$c(I) = 2(n c_e + c_d + c_t + c_v),$$ (4.23)

where $n, c_e, c_d, \text{ and } c_t$ have the same definitions as (4.22), while $c_e$ is the cost to compute, generate and verify a set of parameters for the direct SA.

Now, we already obtain and evaluate the critical part of the proposed SA control scheme between networks. Next, we introduce the mobile-to-net authentication with threshold-based SA to improve the system performance.
4.2.3 Mobile-to-Net Authentication with Threshold-Based Security Association

In this section, we present a threshold-based control scheme for efficient authentication to improve the composite system performance. First, we demonstrate the procedure of proposed scheme on an LAC. Second, since the threshold time to keep a dynamic SA becomes the key of this scheme, we present a utility function with pricing models in which bandwidth efficiency, risk assessment, and number of SAs are considered to derive the optimal threshold. By maximizing the utility function, the optimal threshold can be determined to maximize the composite system performance.

Procedure of the Control Scheme

In the proposed scheme, we only focus on the authentication for inter-domain roaming MNs.

As for inter-domain authentication for visiting MNs, the LAC needs to process two types of dynamic SAs. One is the dynamic SA between LACs/networks; the other is the dynamic SA between the LAC and visiting MN. Since we have introduced SA management scheme between networks in previous section, we only illustrate the control scheme between MNs and LAC in this section.

The control scheme at an LAC works according to different messages that the LAC receives shown in Fig. 4.20. First, if the message is an authentication approval from an HAS, the control scheme generates and assigns necessary credentials to the dynamic SA with the MN. Second, when the message is a notification of service termination from an MN, the control scheme will call an algorithm to determine the optimal threshold time, $\tau_{th}$, to keep an SA for the MN. Third, if the message is an authentication request, the control scheme goes further and checks whether it is an inter-domain authentication request. Our control scheme asks the LAC to trigger a challenge/response authentication protocol carried on DIAMETER to authenticate the intra-domain roaming MNs if the request is an intra-domain request [14, 66]. In the case that the authentication request comes from a visiting MN, our control scheme will ask the LAC to authenticate the MN with existing dynamic SA between the MN and LAC or forward the credentials of the MN to its HAS for authentication according to the existence of the SA between the MN and the LAC.

From the procedure of the proposed scheme shown in Fig. 4.20, we see that
the threshold time to keep the dynamic SA becomes the key to optimize the composite system performance. In order to derive an optimal threshold time to keep the dynamic SA between an LAC and a visiting MN, we need to understand the relationship between the system performance and the threshold. Therefore, we propose a utility function with pricing models next to evaluate this relationship, and finally derive the optimal threshold to maximize the composite system performance.

In order to derive the optimal threshold time to keep a dynamic SA between an LAC and a visiting MN, our approach is to define a utility function with pricing models. We construct a utility function of bandwidth efficiency with pricing models of risk evaluation and number of SAs between an LAC and visiting MNs, which demonstrate the connection of composite system performance and the threshold time to keep a dynamic
Table 4.5: Definition of Symbols for the Mobile-to-Net Authentication on Proposed New Architecture

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$B_s(m)$</td>
<td>Required bandwidth for the service of MN $m$</td>
</tr>
<tr>
<td>$B_{au}(m)$</td>
<td>Bandwidth idle during the authentication of MN $m$</td>
</tr>
<tr>
<td>$T_s(m)$</td>
<td>Service time of MN $m$</td>
</tr>
<tr>
<td>$\tau_{th}(m)$</td>
<td>Threshold time in which we keep dynamic SA for visiting MN $m$ after the end of service time $T_s(m)$</td>
</tr>
<tr>
<td>$T_{au}(\tau_{th}(m))$</td>
<td>Authentication time of visiting MN $m$ with threshold time $\tau_{th}(m)$ for dynamic SA</td>
</tr>
<tr>
<td>$T_{Sa}$</td>
<td>Authentication time of a visiting MN that shares a dynamic SA with the LAC</td>
</tr>
<tr>
<td>$T_{Na}$</td>
<td>Authentication time of a visiting MN that does not share a dynamic SA with the LAC</td>
</tr>
<tr>
<td>$a(m)$</td>
<td>Arrival rate of authentication requests from visiting MN $m$</td>
</tr>
<tr>
<td>$T_{EM}$</td>
<td>Data encryption time at an MN</td>
</tr>
<tr>
<td>$T_{EA}$</td>
<td>Data encryption time at an AP</td>
</tr>
<tr>
<td>$T_{DL}$</td>
<td>Data decryption time at an LAC</td>
</tr>
<tr>
<td>$T_v$</td>
<td>Data verification time at an LAC</td>
</tr>
<tr>
<td>$T_{EL}$</td>
<td>Data encryption time at an LAC</td>
</tr>
<tr>
<td>$T_{DA}$</td>
<td>Data decryption time at an AP</td>
</tr>
<tr>
<td>$T_{DM}$</td>
<td>Data decryption time at an MN</td>
</tr>
<tr>
<td>$T_{SML}$</td>
<td>Time to establish a dynamic SA between a visiting MN and an LAC</td>
</tr>
<tr>
<td>$T_{t1}$</td>
<td>Total message transmission time in authentication of a visiting MN that shares a dynamic SA with LAC</td>
</tr>
<tr>
<td>$T_{t2}$</td>
<td>Total message transmission time in authentication of a visiting MN that does not share a dynamic SA with LAC</td>
</tr>
<tr>
<td>$T_{pg1}$</td>
<td>Total message propagation time in authentication of a visiting MN that shares a dynamic SA with LAC</td>
</tr>
<tr>
<td>$T_{pg2}$</td>
<td>Total message propagation time in authentication of a visiting MN that does not share a dynamic SA with LAC</td>
</tr>
<tr>
<td>$T_{LH}$</td>
<td>Authentication time between LAC and HAS</td>
</tr>
<tr>
<td>$T_{SLH}$</td>
<td>Time to establish a dynamic SA between LAC and HAS</td>
</tr>
</tbody>
</table>

SA. For convenient analysis, the denotations that will be used next are summarized in TABLE 4.5.

Utility Function of Bandwidth Efficiency

Typically, an increase in bandwidth efficiency can improve many QoS parameters such as packet delay and jitter. Thus, we define $U(f)$ as a strictly increasing and
continuous function:
\[ U(f) = e^{(\beta f)} \]

(4.24)

where \( \beta \) is the coefficient to describe how sensitive the required QoS response to \( f \) and \( f \) is bandwidth efficiency of a visiting MN, which is associated with the threshold time to keep a dynamic SA between the LAC and the visiting MN and can be written as:

\[ f(\tau_{th}(m)) = \frac{B_s(m) \cdot T_s(m)}{B_{au}(m) \cdot T_{au}(\tau_{th}(m)) + B_s(m) \cdot T_s(m)} \]

(4.25)

where \( T_{au}(\tau_{th}(m)) \) is the authentication time of a visiting MN \( m \), which is a function of threshold time \( \tau_{th}(m) \) to keep a dynamic SA. The other denotations in (4.25) can be found in TABLE 4.5. Moreover, let \( F \) be the bandwidth efficiency of \( M_m \) inter-domain roaming MNs, \( F \) can be defined as:

\[ F = \sum_{m=1}^{M_m} \frac{B_s(m) \cdot T_s(m)}{B_{au}(m) \cdot T_{au}(\tau_{th}(m)) + B_s(m) \cdot T_s(m)} \]

(4.26)

which will be used to evaluate the system bandwidth efficiency in our simulations.

From (4.25), we can see that the authentication time of a visiting MN, \( T_{au}(\tau_{th}(m)) \), is related with \( \tau_{th}(m) \). Thus, to figure out the relation between \( f(\tau_{th}(m)) \) and \( \tau_{th}(m) \), we need to know \( T_{au}(\tau_{th}(m)) \) first.

Let \( a(m) \) be the arrival rate of authentication requests from MN \( m \), \( TS_{au} \) be the authentication time when a visiting MN that shares a dynamic SA with the LAC requests authentication, and \( TN_{au} \) be the authentication time when the visiting MN that does not share a dynamic SA with the local LAC requests for authentication. The function \( T_{au}(\tau_{th}(m)) \) becomes a combination of different step functions with respect to \( \tau_{th}(m) \) among three scopes, \([0, t_1)\), \([t_1, t_2)\) and \([t_2, \infty)\), i.e.,

\[
T_{au}(\tau_{th}(m)) = \begin{cases} 
TN_{au}, & \text{if } 0 \leq \tau_{th}(m) < t_1 \\
(TS_{au} + TN_{au})/2, & \text{if } t_1 \leq \tau_{th}(m) < t_2 \\
TS_{au1}, & \text{if } \tau_{th}(m) \geq t_2 \\
0, & \text{Otherwise,}
\end{cases}
\]

(4.27)
\[ t_1 = \frac{1}{a(m)} - TN_{au}(m) - T_s(m), \]
\[ t_2 = \frac{1}{a(m)} - TS_{au}(m) - T_s(m). \]

In order to understand \( TS_{au} \) and \( TN_{au} \) clearly, we go further and evaluate them as:

\[
TS_{au} = TEM + T_{EA} + T_{DL} + TV + T_{EL} + T_{DA} + T_{DM} + T_{p1},
\]

(4.28)

\[
TN_{au} = T_{SML} + TEM + T_{EA} + T_{DL} + TLH + T_{EL} + T_{DA} + T_{DM} + T_{p2},
\]

(4.29)

where all of these time parameters are needed for authentication and can be referred to TABLE 4.5. In particular, \( TLH \) can be further expressed as:

\[
TLH = \begin{cases} 
T_{EL} + T_{DH} + TV & \text{if an SA exists} \\
+ T_{EH} + T_{DL}, & \text{between LAC and HAS} \\
T_{SLH} + T_{EL} + T_{DH} & \text{if no SA exists} \\
+ T_{V} + T_{EH} + T_{DL} & \text{between LAC and HAS.}
\end{cases}
\]

(4.30)

To summarize, we introduce a utility function of bandwidth efficiency with an explicit relationship of \( \tau_{th}(m) \). Note that the increase of the threshold time will increase the risk of a dynamic SA to be attacked as well as the number of SAs over time. Thus, we present pricing models to evaluate the risk of keeping a dynamic SA and the number of dynamic SAs in relation to \( \tau_{th}(m) \).

**Pricing Model 1: Risk Function of a Dynamic SA**

All SAs are subject to the risk of being hacked over time, especially under brute force attack [71]. Thus, a decrease of the threshold time to keep a dynamic SA will reduce the risk that an SA is hacked. However, the short threshold time to keep an SA also reduces the chances of an MN to reuse the SA for authentication in our proposed scheme, which forces the MN to establish a dynamic SA first before authentication. Establishing
a dynamic SA is time-consuming, which degrades the bandwidth efficiency because of link idle during authentication waiting time. Therefore, the risk of a dynamic SA and the bandwidth efficiency become a tradeoff in authentication and the risk function is a part of pricing models.

The risk function of a dynamic SA between MN $m$ and a LAC, denoted as $R(m)$, is defined as the product of the probability of one successful attack over a certain period of time with the risk value of the successful attack, which is shown as:

$$R(m) = P_r(m) \cdot C(m).$$  \hspace{1cm} (4.31)

Here, $P_r(m)$ is the probability that one attack succeeds in hacking the dynamic SA between the MN $m$ and a local LAC, and $C(m)$ is the risk value introduced by a successful attack to the SA. Although specific distribution of the probability of one successful attack is unclear during an authentication procedure, it is reasonable to assume that this probability is related to the lifetime of an SA because the success of a brute force attack is related to the lifetime of an SA [71].

In the proposed control scheme, we assume the probability that an attack succeeds regarding the lifetime of the SA is exponentially distributed and independent of different MNs. Although this is not a realistic assumption, it is easy to explain the impact of lifetime on the probability. Because of the simplicity and effectiveness of exponential distribution, it is widely used in the risk assessment of events [58].

With the assumption, the probability that one successful attack happens during an authentication procedure becomes

$$P_r(m) = P_r(T(m)) = \int_0^{T(m)} \lambda(m)e^{-\lambda(m)t}dt,$$  \hspace{1cm} (4.32)

where $T(m)$ is the lifetime of the dynamic SA between MN $m$ and the LAC, and $\lambda(m)$ is the successful attack rate on this SA. In the proposed control scheme, the lifetime of the SA can be written as:

$$T(m) = T_{au}(\tau_{th}(m)) + T_s(m) + \tau_{th}(m).$$  \hspace{1cm} (4.33)

Then the risk function, $R(m)$, can be expressed as

$$R(m) = C(m)(1 - e^{-\lambda(m)(T_{au}(\tau_{th}(m)) + T_s(m) + \tau_{th}(m))}).$$  \hspace{1cm} (4.34)
Here, we consider $C(m)$ a risk value assigned to a security level. Since the concept of security level is widely used in network, environment and power risk assessment [61, 91], we assign two values at two security levels to $C(m)$. If the security level is high, the risk value is high. When an MN is in its home network, in general, the security level is low, which means the user trusts its surroundings. When an MN is beyond its home network, the security level is high, the risk value is accordingly high because the MN may face more threats such as denial of service due to incompatibility in a visiting network.

**Pricing Model 2: Number of Dynamic SAs for MN $m$**

Reducing the number of SAs can be beneficial to reducing the management effort for these SAs and thus improving the manageability of networks, which is of particular interest to large-scale and distributed wireless networks. However, the small number of SAs over time will make visiting MNs unable to reuse the dynamic SAs for authentication, which forces the MN to establish dynamic SA before authentication in our proposed scheme. The establishment of dynamic SA is time-consuming and will degrade the bandwidth efficiency due to bandwidth idle for authentication waiting time. Therefore, the number of dynamic SAs is also a pricing model to the utility function of bandwidth efficiency, and needs to be evaluated carefully.

In order to evaluate the number of SAs between the LAC and the MN $m$, denoted as $n_c(m)$, we explore a similar concept of number of SAs as $N_N$ in (4.18). $N_N$ is different from $n_c(m)$. The former defines the number of dynamic SAs between networks, while the latter is the number of dynamic SAs between visiting MN $m$ and the LAC. Thus, $n_c(m)$ can be written as:

$$n_c(m) = \lim_{T_p \to \infty} \int_0^{T_p} n_m(t) dt, \tag{4.35}$$

where $n_m(t)$ is the function of the number of dynamic SAs between MN $m$ and LAC with respect to time $t$. According to the distribution of $n_m(t)$ as shown in Fig. 4.21, $n_c(m)$ becomes:

$$n_c(m) = \lim_{T_p \to \infty} \frac{1}{T_p} \sum_{k=1}^{K(m)} (T_{aw}(k) + T_s(k)(m) + T_{th}(k)(m)), \tag{4.36}$$

where the superscript $k$ denotes the $k^{th}$ request from MN $m$, and $K(m)$ is the number of authentication requests from MN $m$ within time $T_p$. Assume that $T_{th}(k)(m) = \tau_{th}(m)$,
Figure 4.21: Existence of Dynamic SA Between Visiting MN $m$ and Local LAC.

Thus, we have obtained the utility function of bandwidth efficiency and pricing models of risk assessment and number of dynamic SAs for a visiting MN, all of which are associated with a threshold time $\tau_{th}(m)$. Next, we need to find out the optimal value of $\tau_{th}(m)$ to optimize the composite system performance with respect to bandwidth efficiency, risk assessment, and the number of dynamic SAs for a visiting MN.

Maximize the Utility Function with Pricing Models

After we establish the utility function with pricing models, we need to find out the optimal value of $\tau_{th}(m)$ that maximizes the utility function of bandwidth efficiency with pricing models. Thus, we define a general function $\Gamma[\tau_{th}(m)]$ as:

$$\Gamma[\tau_{th}(m)] = U(f(m)) - \alpha_1 R(m) - \alpha_2 n_c(m),$$

where $\alpha_1 \geq 0$, $\alpha_2 \geq 0$. (4.38)

Here, $\alpha_1$ and $\alpha_2$ are the coefficients of functions $R(m)$ and $n_c(m)$, respectively. Then, the problem to maximize the utility function of bandwidth efficiency with pricing models becomes:

$$\max \quad \Gamma[\tau_{th}(m)]$$

$$\text{over} \quad \tau_{th}(m) \geq 0.$$ (4.39)

If the coefficient $C(m)$ is independent of $\tau_{th}(m)$, then the maximal value
of (4.39) can be obtained at the point of 0, \( t_1 \) or \( t_2 \). That means,

\[
\Gamma_{\text{max}} = \max \{ \Gamma[\tau_{th}(m)]|\tau_{th}(m)=0, \\
\Gamma[\tau_{th}(m)]|\tau_{th}(m)=t_1, \\
\Gamma[\tau_{th}(m)]|\tau_{th}(m)=t_2 \}.
\] (4.40)

After we obtain \( \tau_{th}^*(m) \), we solve the critical part of proposed scheme. Next, we design the algorithm to determine \( \tau_{th}^*(m) \), thus realizing the proposed scheme on an LAC.

**Algorithm to Determine Optimal Time Threshold, \( \tau_{th}(m) \)**

We present an algorithm, shown in Fig. 4.22, to determine \( \tau_{th}^* \). The first part is to initialize the parameters for the inter-domain roaming MN that visits at the first time. The functions \( \text{Measure} \_T_{SML}(MNID) \), \( \text{Measure} \_T_{Na}(MNID) \) and \( \text{Measure} \_T_{LH}(MNID) \) are used to measure different components for the progressive authentication.

The second part implements the utility function to decide the threshold time for the inter-domain roaming MNs that visit for multiple times. The functions, \( \text{Cal} \_\text{Mean} \_T_{Na}(MNID) \) and \( \text{Cal} \_\text{Mean} \_T_{Sa}(MNID) \) are used to calculate the mean time of \( T_{Na} \) and \( T_{Sa} \) based on the current measurement and the past authentication data of the MN. The function \( \text{Cal} \_\text{Auth} \_\text{Arrival} \_\text{Rate}(MNID) \) is applied to calculate the arrival rate of authentication requests for one MN. It is also based on the past number of arrivals of the authentication requests from the MN. The function \( \Gamma[\cdot] \) is the utility function with pricing models defined in (4.38). As proved in Proposition 4.2, the maximal value of \( \Gamma[\tau_{th}] \) can be found at three points of 0, \( t_1 \) or \( t_2 \). As shown in Fig. 4.22, \( \tau_{th}^* \) is obtained by comparing the values of function \( \Gamma[\tau_{th}] \) at these three points. Therefore, the result of \( \tau_{th} \) carried out by this algorithm will be the optimal value to maximize the utility function with pricing models.

### 4.2.4 Simulation

We have conducted simulations for the control scheme based on the new authentication architecture. The purpose of our simulation is to evaluate authentication latency, bandwidth efficiency and number of SAs of our proposed scheme in comparison with existing schemes in distributed and hierarchical authentication architectures.

**Simulation Scenarios and Assumptions**
if \( \text{Re\_Auth\_Num}(MN\_ID) == 1 \)

\[
\begin{align*}
T_{SML}(MN\_ID) &= \text{Measure}_T SML(MN\_ID) \\
T_{N\_au}(MN\_ID) &= \text{Measure}_T T_{N\_au}(MN\_ID) \\
T_{LH}(MN\_ID) &= \text{Measure}_T T_{LH}(MN\_ID) \\
T_{S\_au}(MN\_ID) &= T_{N\_au}(MN\_ID) - T_{LH}(MN\_ID) - T_{SML}(MN\_ID) \\
\tau_{th}^*(MN\_ID) &= 0
\end{align*}
\]

else if

\[
\begin{align*}
T_{N\_au}(MN\_ID) &= \text{Cal}_T \text{Mean}_T T_{N\_au}(MN\_ID) \\
T_{S\_au}(MN\_ID) &= \text{Cal}_T \text{Mean}_T T_{S\_au}(MN\_ID) \\
a(MN\_ID) &= \text{Cal}_T \text{Auth\_Arrival\_Rate}(MN\_ID) \\
t_1(MN\_ID) &= 1/a(MN\_ID) - T_{N\_au}(MN\_ID) - T_s(MN\_ID) \\
t_2(MN\_ID) &= 1/a(MN\_ID) - T_{S\_au}(MN\_ID) - T_s(MN\_ID) \\
\text{if } \Gamma(0) \geq \Gamma(t_1(MN\_ID)) &\& \Gamma(0) \geq \Gamma(t_2(MN\_ID)) \\
\tau_{th}^*(MN\_ID) &= 0 \\
\text{if } \Gamma(t_1(MN\_ID)) \geq \Gamma(t_2(MN\_ID)) &\& \Gamma(t_1(MN\_ID)) \geq \Gamma(0) \\
\tau_{th}^*(MN\_ID) &= t_1 \\
\text{if } \Gamma(t_2(MN\_ID)) \geq \Gamma(t_1(MN\_ID)) &\& \Gamma(t_2(MN\_ID)) \geq \Gamma(0) \\
\tau_{th}^*(MN\_ID) &= t_2
\end{align*}
\]

Figure 4.22: Algorithm to Determine Optimal Threshold, \( \tau_{th}^* \).

Since we focus on the inter-domain authentication, we assume that there are two types of inter-domain roaming MNs. One type is the MNs that are roaming from the home network to the foreign network; the other type is the visiting MNs that are staying in the foreign network and generating authentication requests. Fig. 4.23 shows
the simulation model for signal paths in three architectures.

According to this model, we consider five components: an MN in a foreign network, an AP and an AS inside the visiting network, an HAS in the home network, and a PAS between the AS and the HAS. In distributed authentication architecture, inter-domain authentication is processed through the signal path (2) with static SA between AS and HAS. The hierarchical authentication architecture needs at least one PAS between HAS and AS, according to Fig. 4.1. Therefore, signal path (1) is considered to model the authentication process including PAS. Our proposed authentication architecture uses the signal path (3), which is the same as the signal path in distributed authentication architecture. However, the difference is that AS is replaced with LAC, which implements our new control scheme for efficient authentication.

![Simulation Model for Inter-domain Authentication](image)

(1) Signal Path for Hierarchical Authentication Protocol
(2) Signal Path for Distributed Authentication Protocol with Static SA between AS and HAS
(3) Signal Path for Proposed Control Scheme on New Architecture with Dynamic SA between LAC and HAS

Figure 4.23: Simulation Model for Inter-domain Authentication.

Important parameters used in our simulation are shown in TABLE 4.6. In this table, the total bandwidth for the radio channel is assumed to be 3.84Mbps for audio and video services [40]. Bandwidth of service is assumed to be uniformly distributed with the mean of 50kbps. The bandwidth unavailable for data service in authentication for each request, $B_{au}$, is considered equally to bandwidth for data transmission because
the bandwidth is always examined upon a request [81]. The service time, $T_s$, is assumed to be exponentially distributed with the mean of 1 second. The reason to choose a short service time is that our simulation time is set to 3000 seconds due to the complexity of simulation. In order to model the arrival process of authentication requests, we use two types of arrival process for authentication requests: the arrival rate of new inter-domain authentication requests, $A_{MN}$, and the time interval between successive authentication requests generated by each MN, $T_{int}$. First, $A_{MN}$ is generated with a Poisson process. Then when a new MN is admitted into the system, several authentication requests may be sequentially produced with time interval exponentially distributed around $T_{int}$ by the MN. The time interval between two successive authentication requests is defined as $T_{int} = 1/a - T_{au} - T_s$ according to Fig. 4.21.

We apply a four-way handshake protocol to establish the SA and explore the encryption/decryption cryptographic methods with MD5 and DES. The time for establishing SAs and credential encryption/decryption is obtained from literature [28, 29]. The HAS verification time, $T_v$, is uniformly distributed with the average of 0.1 second. We consider that authentication and registration are completed in the same procedure. Therefore, verification time $T_v$ is the sum of registration time and data decryption/encryption time [29]. The average transfer delay of messages at MAC layer between MN and AR is assumed to have maximum 0.01 second [64]. When the maximum authentication message size is 4096 bytes [14], the transfer delays for the signal path (1) and (2) are defined as 0.07 and 0.05 seconds with the assumption of 2 Mbps link capacity [29].

In order to evaluate the utility function in the simulation, we also define parameters for the $\Gamma [\cdot]$ function in (4.38). In TABLE 4.6, Since an SA in wired equivalent privacy (WEP) can be hacked within as short as 10 minutes [77], we use 10 minutes as the value of $1/\lambda$.

**Simulation Results**

We define the average authentication latency, $T_{av}$, as the ratio of the sum of all the authentication latencies of inter-domain roaming MNs over the number of MNs. We evaluate the $T_{av}$ by changing $A_{MN}$ in Fig. 4.24. We can observe that, the average authentication latency, $T_{av}$, in proposed architecture is far less than that in distributed and hierarchical architectures although it is increasing slowly in the proposed scheme with the increasing arrival rate of inter-domain authentication requests. Compared with
Table 4.6: Simulation Parameters for the Mobile-to-Net Authentication on Proposed New Architecture.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Radio Channel BW</td>
<td>3840 (kbps)</td>
</tr>
<tr>
<td>$B_s$</td>
<td>50 (kbps)</td>
</tr>
<tr>
<td>$B_{au}$</td>
<td>$B_s$</td>
</tr>
<tr>
<td>$T_s$</td>
<td>1 (sec)</td>
</tr>
<tr>
<td>$A_{MN}$</td>
<td>0.2 (calls/sec)</td>
</tr>
<tr>
<td>$T_{int}$</td>
<td>10 (sec)</td>
</tr>
<tr>
<td>MAC Access Delay Time</td>
<td>0.01 (sec)</td>
</tr>
<tr>
<td>Message Transfer Time for Signal Path (1)</td>
<td>0.07 (sec)</td>
</tr>
<tr>
<td>Message Transfer Time for Signal Path (2) (3)</td>
<td>0.05 (sec)</td>
</tr>
<tr>
<td>$T_v$</td>
<td>0.1 (sec)</td>
</tr>
<tr>
<td>$\lambda$ (sec$^{-1}$)</td>
<td>$\frac{1}{600}$</td>
</tr>
<tr>
<td>$\beta$</td>
<td>50</td>
</tr>
<tr>
<td>$\alpha_1$</td>
<td>0.1</td>
</tr>
<tr>
<td>$C$</td>
<td>5</td>
</tr>
<tr>
<td>$\alpha_2$</td>
<td>0.5</td>
</tr>
</tbody>
</table>

In a distributed and hierarchical architecture, the improvement of $T_{av}$ with control scheme in proposed architecture is up to 30% and 34%, respectively. The benefit comes from the control on the establishment of dynamic SAs for visiting MNs. So part of them can reuse a dynamic SA shared with the LAC for authentication. Furthermore, the increasing arrival rate of inter-domain authentication requests reduces the authentication time spent between the LAC and HAS, because more authentication requests can share an SA simultaneously at the LAC, thus reducing the average time to establishing dynamic SA.

As we know, the benefit of bandwidth efficiency in our proposed scheme is gained because some of visiting MNs, previously authenticated in the visiting network, do not need a long time to establish the SA during authentication. However, when the arrival rate of inter-domain authentication requests increases, the number of new visiting MNs is increased. They cannot receive the benefit from the reusing of dynamic SA between an MN and the LAC in the control scheme. Therefore, the bandwidth efficiency in our proposed scheme decreases with the increasing arrival rate of authentication requests in Fig. 4.25, while the bandwidth efficiencies in distributed and hierarchical
Figure 4.24: Authentication Latency vs. Arrival Rate of Authentication Requests.

Figure 4.25: Bandwidth Efficiency vs. Arrival Rate of Authentication Requests.
architectures do not change. However, the bandwidth efficiency with proposed scheme in new architecture is still up to 7.7% and 11% more than that in distributed and hierarchical architecture, respectively.

![Graph showing Average Number of SAs vs. Arrival Rate of Authentication Requests.](image)

Figure 4.26: Average Number of SAs vs. Arrival Rate of Authentication Requests.

In our proposed scheme, the increased arrival rate of inter-domain authentication request causes more authentication requests to share a dynamic SA between the LAC and HAS for authentication. More share of the dynamic SA causes more overlap of existing time of dynamic SAs, which reduces the integration of number of SAs over time. Therefore, the number of SAs between networks is decreasing with the increase of arrival rate of authentication requests, which is shown in Fig. 4.26. As we can see, the improvement of number of SAs with control scheme in proposed architecture is up to 90% and 55% by comparing to the distributed and hierarchical architecture, respectively.

To observe the influence of data service time on the bandwidth efficiency and average number of SAs in our proposed scheme, we change the average session service time in our simulation from 1 second to 19 seconds. From Fig. 4.27, we can see that all of the bandwidth efficiencies in three architectures increase with the increasing of service time, and the improvement of bandwidth efficiency with our scheme on pro-
posed architecture is up to 5% compared to distributed and hierarchical architectures. This increase is due to an extended data service time that can utilize the bandwidth efficiently, which partially reduces the side effect of authentication latency on the bandwidth efficiency. However, although all the bandwidth efficiencies in three models will achieve the same level with the increasing of data service time, our proposed scheme can reach the stable state more quickly than the other two architectures due to the reduction of authentication latency.

At the same time, the increasing data service time decreases the time interval between the end of service time and the start of next authentication request when the arrival rate of authentication requests is fixed. This squeezed time allows that even a small threshold time $\tau_{th}$ can make an MN reuse the dynamic SA with LAC, which reduces the chances for the LAC to establish a dynamic SA with HAS. Thus, the number of SAs between an LAC and an HAS is decreasing, which is shown in Fig. 4.28. The improvement of number of SAs with our scheme in proposed architecture is up to 75% and 88% by comparing to distributed and hierarchical architectures, respectively.

Figure 4.27: Bandwidth Efficiency vs. Average Service Time.
4.3 Conclusions

In this chapter, we propose authentication protocols to improve net-to-net and mobile-to-net authentication efficiency on hierarchical authentication architecture (HAA), because HAA is the most widely used in single-hop wireless networks. The net-to-net authentication is based on a control of traffic based security association (SA), which can be established based on the traffic pattern of MNs. The mobile-to-net authentication is enhanced with two controlled schemes. One is able to establish a local SA based on the signaling cost function for efficient authentication; the other can also achieve authentication efficiency with the establishment of local SA based on a unified function of signaling and risk. Thus, the latter mobile-to-net authentication control scheme not only improves the authentication efficiency, but also improves the security by breaking the local SA timely to avoid potential attacks. The numerical results reveal that these solutions can make significant improvement on the HAA used in single-hop wireless networks.

Furthermore, considering that HAA is not sufficient to network manageability and security due to the use of hop-by-hop static SAs between networks, we propose a
dynamic authentication architecture with two control schemes for net-to-net and mobile-
to-net authentication. The architecture is composed of licensed authentication centers
(LACs), which can create and modify SAs on demand with intelligent control schemes.
In the control schemes, more system parameters are taken into account by using a utility
function of bandwidth efficiency as well as the pricing models of risk assessment and
number of SAs. The simulation results demonstrate that the proposed authentication
architecture and control schemes can improve the manageability and security of networks
by reducing the number of SAs. At the same time, the bandwidth efficiency is enhanced
and the authentication delay is decreased greatly.
Chapter 5

Design and Analysis of

Authentication Mechanisms in

Multi-Hop Wireless Networks

In this chapter, we introduce our work to improve the service availability, which is able to cooperate with proposed authentication protocols for secure communications, in multi-hop wireless networks. Specially, we focus on the clustering system model and formulate the problems on the services availability and communication security in MANet first. Then, since the service availability is an important concern of delivering reliable services, we propose new clustering algorithms to enhance the robustness of hierarchical network architecture in MANet. Third, By applying flexible authentication protocols on the hierarchical clusters, the construction of the clustering is secured, and the security of communications is guaranteed.
5.1 System Model and Problem Formulation

In this section, we first describe a system model on which we are proposing the clustering algorithm for hierarchical architecture. Then, we formulate the problems on the hierarchical clusters in the second part of this section.

5.1.1 System Model

The system model on which our approach is working is composed of a gateway node (GN) and MNs. A GN is a stationary gateway point that can provide Internet access for MNs. We assume that the nodes in the MANet can be either stationary or mobile, which depends on their mobility models. And, all the nodes may have different power levels, which is related with their energy models. Every node needs to join in a cluster to obtain services, in which it can either play a role of cluster head (CH) or act as a cluster member (CM). In order to describe the system model clearly, we define connection node, cluster and adjacent clusters as follows:

Cluster: A cluster is a group of nodes with one of them elected as a head and the other nodes in this group are one hop away from the CH.

Adjacent clusters (ACs): Two clusters are adjacent clusters if there exist at least one path between them by which one cluster can communicate with the other cluster without going through any nodes in third cluster.

Connection node (CN): A node is called connection node if at least one of its neighbor nodes is in different cluster from the cluster to which the node belongs.

Furthermore, we denote a node as \( N_{ih/c}^{(\alpha, \beta)} \), where \( \alpha \) denotes the level of cluster to which the node belongs, \( \beta \) is the sequence number of the cluster on level \( \alpha \), and \( i \) is the identification of the node. \( h/c \) in the superscript is an option to show the role of the node. If the node is a CH, \( h \) is used as the superscript; otherwise, \( c \) is used as the superscript to indicate that the node is a CM. If a set of nodes are grouped in the cluster, we denote the cluster with \( S(\alpha, \beta) \), which is defined as:

\[
S(\alpha, \beta) = [N_i^{h}(\alpha, \beta), N_{2}^{h}(\alpha, \beta), \cdots, N_{m}^{h}(\alpha, \beta)],
\]

where \( m \) is the number of nodes in this cluster, \( N_{i1}^{h}(\alpha, \beta) \) is the CH of this cluster, and the rest of the nodes in this cluster are CMs.
Based on the definitions described above, an example of the system model is shown in Fig. 5.1 with four hierarchical clusters there. Cluster $S(1, 1)$ is composed of four nodes with a CH $N^h(1, 1)$ and three CMs, i.e., $N^c_1(1, 1)$, $N^c_2(1, 1)$, and $N^c_4(1, 1)$. The subscripts 1, 2, 3, and 4 are the identifications of these four nodes. The superscripts $h$ and $c$ indicate the roles of the nodes as CH and CM, respectively. The other nodes in the clusters have the similar denotations as the cluster $S(1, 1)$. In this system model, $S(1, 1)$ and $S(1, 2)$ could be the upper level cluster of $S(2, 1)$, which depends on the service availability of these two clusters in our proposed algorithm. $S(2, 1)$ is the upper level cluster of $S(3, 1)$ since $S(2, 1)$ is the only cluster that provides connectivity for $S(3, 1)$ to the hierarchical clusters.

The system model is very close to the reality and easy to deploy due to the following reasons:

- Gateway nodes have been applied in many wireless networks such as wireless local area network (WLAN), mobile IP network, and mesh network. Thus, the system model of MANet in our method can be viewed as an extension of WLAN, mobile IP network, and mesh network.

- The clustering technique has been proposed and developed by many papers in wireless networks [10, 42, 56, 78, 79], which provides strong support to communications in MANet.
Next, we formulate the problems that are addressed in this chapter based on the model.

5.1.2 Problem Formulation

Since the objective of our approach is to improve the service availability and security in MANet, the most important thing is to find out what exactly the service availability is in MANet. After that, the related factors that affect the service availability and security are discussed to clarify the problems.

Definition of Service Availability

_Service availability_ can be defined as the probability of the system being found in the operating state at some time _t_ in the future given that the system started in the operating state at time _t = 0_[19, 27]. Quantitatively, _service availability_ can be defined as the ratio of mean time to failure (MTTF) over the sum of MTTF and mean time to recover (MTTR). Although this definition is good to measure the performance of a system with central servers, this definition is not suitable for the clustering in MANet due to the following reasons:

- Central servers do not exist in MANet. Instead, nodes with weak ability are serving for each other. A failure of one node is unlikely to result in tremendous failures.
- The node properties induce more factors causing failures that cannot be recovered. For example, the energy of a node may not satisfy the requirement of services even though it can be recharged because the speed of energy recharge is very slow and depends on the appearance of events, and the movement of a node may result in the disruption of communication sessions immediately.

In order to evaluate the _service availability_ in MANet, we propose two variables to measure _node service availability_, and _system service availability_.

Node Service Availability: In our approach, we consider that a _failure_ happens when an MN in a route fails to deliver data in a communication session due to energy exhaustion or movement out of the transmission coverage of its neighbor nodes. Let the mean
time to failure when the energy exhaustion or node movement out of the transmission coverage of node $i$'s neighbor nodes happens be $MTTF_{e,r}(i)$, the node service availability of node $i$, $R_n(i)$, could be evaluated with $MTTF_{e,r}(i)$. Since $MTTF_{e,r}$ depends on the energy and movement of the node $i$, $R_n(i)$ can be defined as:

$$R_n(i) = MTTF_{e,r}(i) = \min\{MTTF_e(i), MTTF_r(i)\},$$

(5.2)

where $MTTF_e(i)$ is the mean time to failure due to energy depletion of node $i$, $MTTF_r(i)$ is the mean time to failure due to node $i$'s movement out of the transmission overlap of all neighbor nodes.

The $MTTF_{e,r}(i)$ in our approach is different from the traditional definition in wired networks, which is defined as the average time for a system to meet a failure due to many factors, and these failures may be recovered soon for services. The $MTTF_{e,r}(i)$ in our approach, however, is defined for node $i$ with two failure factors, i.e., energy exhaustion and movement out of transmission coverage. And after the failure happens, it is unlikely for the node to recover for services.

Although the failure of a common node may not generate tremendous effects, the failure of a CH could affect many nodes and, even worse, change the architecture of the hierarchical clusters. Thus, it is important to choose the node with big value of $MTTF_{e,r}$ as the CH in the area. This selection should also depend on the connectivity of the nodes in the area. In order to evaluate the service availability of a group of nodes for the other nodes in an area, we introduce a variable system service availability next.

**System Service Availability**: Since CHs in clustering techniques are the most important due to their heavy duties in a certain area, we define a new terminology system service availability in this approach. Let $G(V, E)$ denote a group of nodes in an area with connectivities, where $V$ is the set of nodes in this group and $E$ is the set of edges in this group, system service availability can be defined as the ratio of the sum of MTTF of all CHs in the graph $G(V, E)$ over the number of all CHs in $G(V, E)$. Denote the service availability in a graph as $R_s$, it can be shown as:

$$R_s = \frac{1}{|\Omega_h^{(V)}|} \sum_{i \in \Omega_h^{(V)}} MTTF_{e,r}(i),$$

(5.3)

where $\Omega_h^{(V)}$ is the set of nodes that are CHs in $V$, $|\Omega_h^{(V)}|$ is the number of the nodes in the set $\Omega_h^{(V)}$, $i$ is the identification of the node in the set $\Omega_h$, and $MTTF_{e,r}(i)$ is the mean time to failure of node $i$. 
Problem Statement

Since our purpose is to improve the service availability and security among hierarchical clusters in MANet, we need to maximize the service availability first. Thus, the first problem becomes, given $G(V, E)$, $R_n(j)$, $j \in V$:

$$\text{Maximize } \{R_s\}, \quad (5.4)$$

subject to $|\Omega_h^{(V)}| \geq 0$ in constructing hierarchical clusters.

If $G(V, E)$ denotes the entire MANet, the optimization problem can be implemented in the MANet, and $R_s$ is the network service availability. However, in a MANet of big scale, it is impossible for the entire network to know the status of all nodes and edges. Therefore, it is unrealistic to utilize the information of the entire network to optimize the system performance. Instead, the local groups of nodes should be considered for the local optimization.

Since the problem of the local optimization appears, the size of $G(V, E)$ should be determined to achieve best performance. In [88], the authors prove that the best performance in terms of reliability of routing can be achieved if the number of hops between two nodes is two or three. Since the CHs in hierarchical clusters could take care of routing, the transmission reliability between them should be guaranteed. Therefore, we use one-hop cluster in proposed clustering algorithms, which can guarantee that the biggest number of hops between two CHs is three.

In order to clarify the problem of local optimization, we want to use a denotation $G(V_i, E_i)$ to represent the group of nodes centered and connected with node $i$. $V_i$ is the set of nodes that include node $i$ and the neighbor nodes around node $i$, and $E_i$ is the set of edges connected with node $i$. Thus, the problem in (5.4) can be changed as follows:

**Problem of Constructing Hierarchical Clusters for Local Optimization:**

Given $G(V_i, E_i)$, $R_n(j)$, $j \in V_i$

$$\text{Maximize } \{R_s\}, \quad (5.5)$$

subject to $|\Omega_h^{(V_i)}| \geq 0$.

From the definitions described in (5.2) and (5.3), we can see that $R_n(i)$ and $R_s$ are all related to $MTTF_{e,r}(i)$. For $MTTF_{e,r}(i)$, when a node uses up its energy or moves
out of a cluster, it cannot provide the service for any nodes in the cluster. Furthermore, the architecture of the cluster may be ruined due to the loss of the connectivity provided by the node. Therefore, the evaluation of $MTTF_{e,r}(i)$ is very important to construct and maintain the architecture of hierarchical clusters and is provided in Section 5.2.1.

After we evaluate $MTTF_{e,r}(i)$, in order to provide a robust platform to improve service availability and security, several algorithms are needed to construct hierarchical clusters securely and handle node failures with respect to the optimization of the service availability $R_s$. During this procedure, we need to consider the following aspects:

- The optimization is associated with the number of CHs in the set of $\Omega_h^{V_i}$, which is determined by the algorithms. After we find out the optimal number of $|\Omega_h^{V_i}|$ in the algorithms, the optimization of $R_s$ is achieved with these algorithms. We develop these algorithms in Section 5.2.2.

- Constructing hierarchical clusters securely should be entangled with the authentication mechanisms to guarantee security. Thus, strong authentication protocols in different cases in the construction processes should be developed. We propose these authentication protocols in different cases in Section 5.3.

Therefore, we introduce the robust clustering algorithms to improve the service availability first, followed by the flexible authentication protocols in different cases, which are involved with the construction processes for security.

5.2 Robust Clustering Algorithms to Improve the Service Availability

In order to improve the service availability in MANet, we need to know how to evaluate the service availability in MANet. Therefore in this section, we first provide a method for the evaluation of service availability. Based on the evaluation, we propose robust clustering algorithms to improve the service availability.
5.2.1 Evaluation of $MTTF_{e,r}(i)$

As mentioned before, $MTTF_{e,r}(i)$ is the mean time to failure of node $i$ due to energy depletion or node movement out of transmission coverage. Thus, in order to evaluate $MTTF_{e,r}(i)$, $MTTF_e(i)$ and $MTTF_r(i)$ need to be evaluated.

**Evaluation of $MTTF_e(i)$**

We consider the energy consumption of a node when transmitting and receiving packets on medium access control (MAC) layer since the transmission and reception of data in MANet are the biggest energy consumption [39]. Although some energy models are proposed to evaluate the optimal energy consumption [41, 85], they do not consider energy replenishment.

In our analysis, we use an energy model on the MAC layer, which considers the energy replenishment and is similar as the energy model developed in [57]. The difference is that we introduce some random events such as sunlight and vibration to explain the replenishment of energy, which is more realistic in practical environments. The energy model in our analysis can be defined as:

$$E(i, \tau) = \min\{E'(i, \tau) + \sum_{k=1}^{N_e(i, \tau)} \gamma \cdot t_k(i, \tau), \ E_{\max}\} - [\varepsilon_r \cdot \eta + \varepsilon_s \cdot (1 - \eta)] \cdot N_p(i, \tau),$$

(5.6)

where $i$ is the identification of an MN, $\tau$ is a a period of time, $E'(i, \tau)$ is the energy that MN $i$ has at the end of time period $\tau$, $E'(i, \tau)$ is the energy that MN $i$ has at the beginning of time period $\tau$, $N_e(i, \tau)$ is the number of the events for energy replenishment of node $i$ that happen in time $\tau$, $\gamma$ is the speed to charge the energy, $t_k(i, \tau)$ is the duration time of the energy replenishment when $k^{th}$ event of replenishment happens for node $i$ in time $\tau$, $E_{\max}$ is the maximal energy that the MN can have, $\varepsilon_r$ is the energy to receive a packet, $\varepsilon_s$ is the energy to send a packet, $N_p(i, \tau)$ is the number of packets relayed by MN $i$ in time $\tau$, and $\eta$ is the ratio of the received packets.

Furthermore, the number of packets relayed by MN $i$ in time $\tau$, i.e., $N_p(i, \tau)$, can be determined by:

$$N_p(i, \tau) = F(i, \tau) \cdot \tau,$$

(5.7)

where $F(i, \tau)$ is the throughput of node $i$ in time $\tau$. Moreover, if the energy of a node
i after time $\tau$, i.e., $\delta(i, \tau)$ is less than a threshold energy $\delta_{th}(i)$, node $i$ fails to serve for other nodes. Therefore, the probability that node $i$ can serve for its neighbor nodes with enough energy in time $\tau$, i.e., $P_s(i, \tau)$, can be written as:

$$P_s(i, \tau) = P_{E\{E(i, \tau) > \delta_{th}(i)\}}, \quad (5.8)$$

where $\delta(i, \tau)$ is defined in (5.6). Then, $MTTF_e(i)$ can be determined by:

$$MTTF_e(i) = \int_0^\infty \tau \cdot P_s(i, \tau) d\tau. \quad (5.9)$$

In order to calculate $P_s(i, \tau)$, we need to derive the PDF of random variable $\delta(i, \tau)$ defined in (5.6). To derive the PDF of $\delta(i, \tau)$, we define variables $X_j$, ($j=1, 2, 3, 4$), as:

$$X_1 = \sum_{k=1}^{N_e(i, \tau)} t_k(i, \tau), \quad X_2 = \gamma X_1, \quad X_3 = \delta'(i, \tau) + X_2, \quad X_4 = \min\{X_3, \delta'_{max}\}, \quad (5.10)$$

where $X_1$ is the real energy charging time in time $\tau$, $X_2$ is the theoretically charged energy, $X_3$ is the theoretical energy level, and $X_4$ is the real energy level. To derive the PDF of $E(i, \tau)$ becomes to find out the PDFs of $X_j$, ($j=1, 2, 3, 4$), subsequently.

Let $f_{X_1}(t)$ be the PDF of $X_1$, it can be written as:

$$f_{X_1}(t) = \sum_{k=1}^{\infty} f(t, k) = \sum_{k=1}^{\infty} f(t/k) \cdot f_{N_e(i, \tau)}(k), \quad (5.11)$$

where $f(t, k)$ is the PDF of the energy charging time when $k$ events that can charge the battery happen, $f(t/k)$ is the conditional PDF of the energy charging time given $k$ energy charging events happen, and $f_{N_e(i, \tau)}(k)$ is the PDF of random variable $N_e(i, \tau)$.

In (5.6), we only consider $N_e(i, \tau)$ and $t_k(i, \tau)$ the random variables. As for $F(i, \tau)$, in worst case, in which the neighbor nodes of node $i$ are trying to send as many packets as possible to node $i$, and the channel condition and the number of neighbor nodes of node $i$ are fixed, the ideal maximal throughput $F$ can be determined, which is unrelated with the time $\tau$ [8, 33]. Thus, the mean value of ideal maximal throughput $F$ can be carried out by node $i$ with the number of its NNs.

We assume $N_e(i, \tau)$ is Poisson distributed with mean value $\lambda_n \tau$ and $t_k(i, \tau)$ is independent, identical exponentially distributed with mean value $1/\lambda_t$, i.e.,

$$f_{N_e(i, \tau)}(k) = e^{-\lambda_n \tau} \frac{[\lambda_n \tau]^k}{k!}, \quad (5.12)$$

and $f_{t_k}(t) = \lambda_t e^{-\lambda_t t}, \quad (5.13)$
where $f_{N_e(i, \tau)}(k)$ is the PDF of random variable $N_e(i, \tau)$, $f_{t_k}(t)$ is the PDF of random variable $t_k(i, \tau)$.

Because $f(t/k)$ is the conditional PDF of the energy charging time given $k$ energy charging events happen, the energy charging time in this case could be written as:

$$t = \sum_{k=1}^{K} t_k(i, \tau). \quad (5.14)$$

Since the PDF of $t_k(i, \tau)$ is exponential distribution, $f(t/k)$ can be determined by:

$$f(t/k) = \frac{\lambda t(t-1)}{(k-1)!} \lambda^t e^{\lambda t}, \quad t \geq 0, \quad (5.15)$$

By substituting (5.12) and (5.15) into (5.11), we can calculate $f_X(t)$. The PDFs for $X_2$, $X_3$, and $X_4$ can be carried out based on $f_X(t)$ as:

$$f_{X_2}(t) = \frac{1}{\tau} f_X(t),$$

$$f_{X_3}(t) = f_{X_2}(t - \delta(i, \tau - 1)),$$

$$f_{X_4}(t) = \begin{cases} f_{X_3}(t) & 0 \leq t \leq \varsigma_{max} \\ 0 & t > \varsigma_{max} \end{cases}, \quad (5.16)$$

Thus, if we define $\delta = [\varepsilon_r \cdot \eta + \varepsilon_s \cdot (1 - \eta)] \cdot \tau \cdot F(i, \tau)$, $MFFT_e(i)$ is determined by:

$$MFFT_e(i) = \int_0^\infty \tau \cdot P_s(i, \tau) d\tau$$

$$= \int_0^\infty \tau \cdot P_x \{X_4 > \delta \} d\tau$$

$$= \int_0^\infty \tau \cdot P_x \{X_4 > \delta + \delta_{th(i)}\} d\tau$$

$$= \int_0^\infty \tau \cdot \left[ \int_{\delta + \delta_{th(i)}}^{\infty} f_{X_4}(t) dt \right] d\tau, \quad (5.17)$$

where $f_{X_4}(t)$ is defined in (5.16).

**Evaluation of $MTTF_r(i)$**

$MTTF_r(i)$ depends on the movement and mobility model of node $i$. Since random way-point mobility model is widely used in MANet for simulations, we use it to estimate $MTTF_r(i)$. In random way-point mobility model, an MN begins by staying in one location for a certain period of time, i.e., a pause time, which is uniformly distributed
between 0 and a maximal time $T_{pm}$. Once this time expires, the node chooses a random destination in the simulation area and a speed that is uniformly distributed between minimal speed $c_{min}$ and maximal speed $c_{max}$. The node then travels toward the newly chosen destination at the selected speed. Once the node pauses at the new point, this process is repeated [38].

Thus, the basic mobility model in our analysis can be described with pause time $T_p$, speed $C$, and destination $D$ of an MN. Let $f_{T_p}(t)$, $f_C(c)$, and $f_D(d)$ be the PDFs of $T_p$, $C$, and $D$, respectively. Then, they can be written as:

$$f_{T_p}(t) = \begin{cases} \frac{1}{T_{pm}}, & t \in [0,T_{pm}] \\ 0, & \text{else} \end{cases},$$

$$f_C(c) = \begin{cases} \frac{1}{c_{max} - c_{min}}, & c \in [c_{min},c_{max}] \\ 0, & \text{else} \end{cases},$$

$$f_D(d) = \begin{cases} \frac{1}{N_A}, & d \in [0,N_A] \\ 0, & \text{else} \end{cases}.$$  

(5.18)

where $N_A$ is the number of points that can be selected in the area. In continuous region, $N_A$ becomes the area of the region.

In order to evaluate $MTTF_r(i)$, let us consider a scenario shown in Fig. 5.2. In this figure, node $i$ is in the transmission overlay of node 1, 2, and 3. If node $i$ stays in the shadow area and there are no changes of the locations of node 1, 2, and 3, node $i$ can keep serving them when node 1, 2 or 3 ask for relaying data to node $i$. However, if node $i$ moves out of the shadow area, it will not be able to serve these three nodes simultaneously. Thus, evaluation of $MTTF_r(i)$ is to evaluate the mean time that node $i$ stays in the shadow area.

Let $T$ be the mean time that node $i$ stays in the shadow area in Fig. 5.2, it can be calculated as:

$$T = T_p + \sum_{n=1}^{\infty} (T_m + T_p)P^n = T_p + (T_m + T_p)\frac{P}{1-P},$$  

(5.19)

where $T_p$ is a pause time of node $i$ at the point, $T_m$ is the movement time of node $i$ between two points, $P$ is the probability that node $i$ stays in the shadow area after one movement, and $P^n$ is the probability that node $i$ stays in the shadow area after
$n^{th}$ movement. Since the selection of next moving point in random way-point mobility model is an independent, identical and uniform distribution in the area $A$, which is shown in Fig. 5.2, $P$ can be determined by:

$$P = \frac{S_s}{S_A},$$ (5.20)

where $S_A$ is the area of the MANet, and $S_s$ is the shadow area.

In reality, although $S_A$ could be estimated by the coverage of the MANet, the estimation of $S_s$ is difficult to be obtained because the overlapped coverage is caused by many neighbor nodes around node $i$.

In order to evaluate the overlapped transmission coverage, we propose a terminology service guarantee area, which defines the minimal area in which an MN can provide service for all of its neighbors, assuming that its neighbors are stationary. Then, if we denote a service guarantee area as $S_g$, it can be calculated as:

$$S_g = \pi \cdot r_g^2, \text{ where } r_g = \min\{r - d_{i,j}\} \quad j \in V_i.$$ (5.21)
In (5.21), \( j \) is the identification of a neighbor node in the set of \( V_i \), and \( d_{i,j} \) is the distance between node \( i \) and \( j \). \( d_{i,j} \) can be carried out with the channel propagation model as well as the signal power that the node \( i \) receives [62]. As shown in Fig. 5.2, the smallest gray circle around node \( i \) is the service guarantee area. If we calculate the area of the MANet, i.e., \( S_A \), according to Fig. 5.2, the probability that node \( i \) stays in the service guarantee area after one movement, i.e., \( P_g \), is

\[
P_g = \frac{S_g}{S_A} = \frac{\pi \cdot r_g^2}{a^2}.
\]

Thus, mean value of \( \bar{T} \) in the service guarantee area can be found as:

\[
\bar{T} = \bar{T}_p + (\bar{T}_m + \bar{T}_p) \frac{P_g}{1 - P_g},
\]

where \( P_g \) is shown in (5.22), \( \bar{T}_p = \frac{T_{pm}}{2} \), which can be derived from \( T_p \)'s PDF in (5.18), and \( \bar{T}_m \) is the mean value of movement time \( T_m \), which can be expressed as [7]:

\[
T_m = \bar{D} \log\left(\frac{c_{max}}{c_{min}}\right)/(c_{max} - c_{min}),
\]

where \( \bar{D} \) is the average distance passed by the node, which can be written as [7]:

\[
\bar{D} = \int_0^a \int_0^a \int_0^a \int_0^a \left[(x_2 - x_1)^2 + (y_2 - y_1)^2\right]^{\frac{1}{2}} dx_1 dx_2 dy_1 dy_2 = 0.521405a,
\]

where \( a \) is the length of the square area of the MANet. Thus, \( MTTF_r(i) \) can be obtained as \( MTTF_r(i) = \bar{T} \).

Since we obtain the evaluations of \( MTTF_e(i) \) and \( MTTF_r(i) \), the evaluation of \( MTTF_{e,r}(i) \) could be found through (5.2). In order to achieve maximal service availability \( R_s \), we still need to understand the optimal number of \( |\Omega_h| \), which depends on the algorithms to construct the hierarchical clusters.

Thus, we introduce algorithms next, which are able to construct hierarchical clusters by adding nodes or removing nodes of failures in a hierarchical architecture.

### 5.2.2 Adaptive Algorithms to Construct Hierarchical Clusters

The construction of hierarchical clusters in our approach can be viewed as processes that add nodes to or remove nodes of failures from an existing hierarchical
architecture. These two processes are illustrated next respectively.

**Process to Add a Node into Hierarchical Clusters**

The process to add a node to an existing hierarchical architecture can be implemented with two algorithms:

- When a node wants to join in existing hierarchical clusters, it differentiates its neighbor nodes into different groups according to their roles.
- The joining node chooses a neighbor node to connect according to the roles of the neighbor nodes and service availabilities.

In the first algorithm, the joining node can differentiate its neighbor nodes according to their roles included in their messages. The second algorithm is more complicated than the first one because the joining node needs to decide its role and connectivity to one of its neighbor nodes, so that the hierarchical clusters are formed and the system service availability is maximized.

**Classification Algorithm**

In the classification algorithm shown in Fig. 5.3, \( N_i \) is a joining node with the knowledge of its neighbor nodes denoted as \( G(V_i, E_i) \). The joining node \( N_i \) sends inquiry messages to the neighbor node \( N_j, j \in V_i \). According to the role and node service availability of the node \( N_j \) included in the replied message, \( N_i \) saves \( N_j \)’s node service availability, and puts it into one of the four sets, i.e., \( V^{(i)}_{CH1}, V^{(i)}_{CH2}, V^{(i)}_{CM}, \) and \( V^{(i)}_{NO} \). The complexity of this algorithm is \( O(|V_i|) \) since it needs to repeat for all \( j \in V_i \). The result after running the algorithm is shown in Fig. 5.4.

**Joining Algorithm**

In the joining algorithm shown in Fig. 5.5, node \( i \) determines its operation according to the information collected from its neighbor nodes.

In case 1, if there are some CHs around node \( i \), node \( i \) compares its service availability with the threshold calculated from the service availabilities of the CHs. When node \( i \)’s service availability is below the threshold, node \( i \) becomes a CM of the CH that has the maximal node service availability in the sets of \( V^{(i)}_{CH1} \) and \( V^{(i)}_{CH2} \). When node \( i \)’s service availability is above the threshold, node \( i \) becomes a CH associated with
the CH with maximal service availability in the sets of $V^{(i)}_{CH1}$ and $V^{(i)}_{CH2}$. The association between them determines $\alpha_i$ and $\beta_i$ for node $i$. The complexity of the algorithm in case 1 is $O(1)$ since node $i$ just needs to update its status based on the threshold value calculated from the sets of values collected from the classification algorithm. The figure after running this algorithm in case 1 can be shown in Fig. 5.6.

In case 2, if there are no CHs around node $i$, it searches for CMs in its neighbor nodes. If node $i$ finds any CMs around itself, it means that node $i$ is able to join the hierarchical clusters. Then, node $i$ becomes a CH. The $\alpha_i$ and $\beta_i$ will be determined

Input:
Given $G(V_i, E_i)$, the sub-graph of neighbor nodes around node $i$, and $R_n(j)$, $j \in V_i$, the node service availability of the nodes in $V_i$; $N_i$ is the joining node $i$.

Output:
$V^{(i)}_{CH1}$, the set of nodes that are CHs with CMs around node $i$;
$V^{(i)}_{CH2}$, the set of nodes that are CHs without CMs around node $i$;
$V^{(i)}_{CM}$, the set of nodes that are CMs around node $i$;
$V^{(i)}_{NO}$, the set of nodes that are neither CHs nor CMs around node $i$;
$R_n(V_i)$, the set of node service availabilities owned by the set of nodes $V_i$;

Step.1 $N_i$: Sends inquiry message to node $N_j$, $j \in V_i$
Step.2 $N_i$: On receiving the replied message from $N_j$, saves $N_j$’s node service availability.
$R_n(V_i) \leftarrow R_n(j)$;
Step.3 $N_i$: Checks the role of $N_j$, $j \in V_i$, and put it in the set according to its role.
Case 1: $N_j$ is a CH with CMs: $V^{(i)}_{CH1} \leftarrow N_j$
Case 2: $N_j$ is a CH without CMs: $V^{(i)}_{CH2} \leftarrow N_j$
Case 3: $N_j$ is a CM: $V^{(i)}_{CM} \leftarrow N_j$
Case 4: $N_j$ is neither a CH nor a CM: $V^{(i)}_{NO} \leftarrow N_j$
Step.4 Repeat steps 1,2, and 3 for all $j \in V_i$.

Figure 5.3: Classification Algorithm
by the CH of the CM that has biggest value of node service availability in the set of $V_{CM}(i)$. Same as case 1, the complexity of the algorithm in case 2 is $O(1)$. The result after running the joining algorithm in this case is shown in Fig. 5.7.

In case 3, node $i$ cannot find any neighbor nodes that are connected with the hierarchical clusters. Therefore, node $i$ just goes back to classification algorithm and keeps inquiring its neighbor nodes periodically. The complexity of the algorithm in case 3 is $O(|V_i|)$, the same as the classification algorithm.

After running the classification and joining algorithms, node $i$ joins in the hierarchical architecture if at least one of its neighbors is in the architecture.

After we finish the process to add a node into the hierarchical clusters, we still need to process the removal of a node from the architecture if a failure of a node happens, which is introduced next.
Input:
Given $V_{CH1}^{(i)}, V_{CH2}^{(i)}, V_{CM}^{(i)}, V_{NO}^{(i)}$, and $R_n(j), j \in V_i$.

Output:
$N_i^c(\alpha_i, \beta_i)$ or $N_i^h(\alpha_i, \beta_i)$
or node $i$ keeps searching.

Case 1 $|V_{CH1}^{(i)}| + |V_{CH2}^{(i)}| > 0$: Check if
$R_n(i) \leq \sum_{j \in V_{CH1}^{(i)}} R_n(j) /(|V_{CH1}^{(i)}| + |V_{CH2}^{(i)}|)$.
If so, node $i$ becomes $N_i^c(\alpha_i, \beta_i)$; Otherwise, node $i$ becomes $N_i^h(\alpha_i, \beta_i)$, where $\alpha_i = \alpha_j + 1$ and $\beta_i$ is assigned by $N_{jm}^h(\alpha_j, \beta_j)$, $j_m \in V_{CH1}^{(i)}$ or $V_{CH2}^{(i)}$ and $R_n(j_m)$ is maximal.

$S(\alpha_i, \beta_i) = [N_i^h(\alpha_i, \beta_i), N_{i1}^c(\alpha_i, \beta_i), \ldots, N_{im}^c(\alpha_i, \beta_i)]$.

Case 2 $|V_{CH1}^{(i)}| = |V_{CH2}^{(i)}| = 0$ and $|V_{CM}^{(i)}| > 0$: node $i$ becomes $N_i^h(\alpha_i, \beta_i)$, where $\alpha_i$ and $\beta_i$ are calculated and assigned by the CH of $N_{jm}^c(\alpha_j, \beta_j)$, $j_m \in V_{CM}^{(i)}$ and $R_n(j_m)$ is maximal.

Case 3 $|V_{CH1}^{(i)}| = |V_{CH2}^{(i)}| = |V_{CM}^{(i)}| = 0$: node $i$ has no neighbor nodes in the hierarchical clusters. Thus, node $i$ goes back to classification algorithm and keeps inquiring its neighbor nodes periodically for joining.

Figure 5.5: Joining Algorithm

Algorithm to Remove a Node

The process to remove a node of failure from an existing hierarchical architecture is different to CM and CH. Even for CMs, the removal process of a CM that is not a CN is different from the process to remove a CM that is CN, because the removal of a CN is likely to change the architecture of hierarchical clusters.

Then, the process to remove nodes is shown as follows with different operations to CMs, CNs, and CHs:

- If a CM that is not a CN is being removed, the CH of the CM simply removes the record of the CM from the cluster.

- If a CM that is a CN is being removed, the CH of the CM needs to check the connectivity to corresponding cluster that the CN connects besides removing the
Figure 5.6: Joining Algorithm in Case 1.

- If a CH is being removed, the CMs in the cluster need to negotiate and generate a new CH for this cluster. For the CMs that lose connectivities to the new cluster due to the CH failure, adding process is called to put them into different clusters.

According to the descriptions, the removing algorithm is shown in Fig. 5.8.

In case 1, failing node is just a general CM, $N^c_i(\alpha_i, \beta_i)$, that is not a CN. Therefore, the CH of this cluster $N^h_i(\alpha_i, \beta_i)$ simply removes the record of $N^c_i(\alpha_i, \beta_i)$ with $O(1)$ operation.

In case 2, failing node is a CM and also a CN that connects another cluster. Thus, $N^h_i(\alpha_i, \beta_i)$ needs to check the connectivity to the cluster originally through $N^c_i(\alpha_i, \beta_i)$. The searching process is to inquire the rest of the CMs in the cluster $S(\alpha_i, \beta_i)$. Thus, the complexity of the algorithm in case 2 is $O(|S(\alpha_i, \beta_i)|)$, where $|S(\alpha_i, \beta_i)|$ is the number of nodes in the cluster $S(\alpha_i, \beta_i)$. If the connectivity still exists, $N^h_i(\alpha_i, \beta_i)$ just removes $N^c_i(\alpha_i, \beta_i)$ with $O(1)$ operation. If the connectivity is broken, $N^h_i(\alpha_i, \beta_i)$ needs to find if it should re-establish the connectivity to disconnected clus-
When the disconnected cluster is in the lower level, i.e., $\alpha_i + 1$, $N_h^{\alpha_i, \beta_i}$ only removes the record of the cluster with $O(1)$ operation. If the disconnected cluster is in the higher level, i.e., $\alpha_i - 1$, $N_h^{\alpha_i, \beta_i}$ needs to find a new attached cluster in the higher level. The searching process starts from $S(\alpha_i, \beta_i)$, and goes through all the clusters at the lower level of $S(\alpha_i, \beta_i)$ to find a cluster with neighbor cluster that is able to bring them back to the hierarchical architecture. If we assume that the maximal level of this hierarchical architecture is $\alpha_{\text{max}}$, the maximal number of clusters at a level is $\beta_{\text{max}}$, and the maximal number of nodes in a cluster is $|S|_{\text{max}}$, the complexity of the removing algorithm in case 2 is $O(\beta_{\text{max}}^{\alpha_{\text{max}}} \cdot |S|_{\text{max}})$ in the worst case since there may be maximal number of $\beta_{\text{max}}^{\alpha_{\text{max}}} \cdot |S|_{\text{max}}$ needed to be searched. Thus, the complexity of the removing algorithm in case 2 is $O(\beta_{\text{max}}^{\alpha_{\text{max}}} \cdot |S|_{\text{max}})$. The result of the removing algorithm in case 2 can be shown in Fig. 5.9.

In case 3, when failing node $i$ is a CH, $N_i^{h}(\alpha_i, \beta_i)$, the negotiation process needs $O(|S(\alpha_i, \beta_i)|)$ operations because the negotiation is only performed among the CMs in the cluster. For the CMs that cannot be connected into the newly constructed cluster
Input:
Given $N^{h/c}_i(\alpha_i, \beta_i)$, the failing node, and $G(V_m, E_m)$, the graph centered and connected with node $m$ in one hop, where $N^{h/c}_m(\alpha_i, \beta_i) \in S(\alpha_i, \beta_i)$.

Output:
Make $N^{h/c}_m(\alpha_i, \beta_i)$ reconnected to hierarchical clusters, where $m \neq i$ and $N^{h/c}_m(\alpha_i, \beta_i) \in S(\alpha_i, \beta_i)$.

Case 1 Failing node $i$ is a CM, $N^c_i(\alpha_i, \beta_i)$, not a CN: $N^h(\alpha_i, \beta_i)$ removes the record of $N^c_i(\alpha_i, \beta_i)$.

Case 2 Failing node $i$ is a CM, $N^c_i(\alpha_i, \beta_i)$, also a CN: $N^h(\alpha_i, \beta_i)$ checks if the connectivity to the correspondent cluster through $N^c_i(\alpha_i, \beta_i)$ exists or not. If the connectivity exists, $N^h(\alpha_i, \beta_i)$ does nothing but removes the record of $N^c_i(\alpha_i, \beta_i)$. If the connectivity does not exist and the disconnected cluster is in the level of $\alpha_i + 1$, $N^h(\alpha_i, \beta_i)$ just removes the records of $N^c_i(\alpha_i, \beta_i)$ and the disconnected cluster. If the connectivity does not exist and the disconnected cluster is in the level of $\alpha_i - 1$, $N^h(\alpha_i, \beta_i)$ tries to find the connected cluster in the higher level among its neighbor clusters.

Case 3 Failing node $i$ is a CH, $N^h_i(\alpha_i, \beta_i)$: $N^c_m(\alpha_i, \beta_i)$, $m \neq i$ and $m \in S(\alpha_i, \beta_i)$, negotiates to construct new cluster, trying to keep the CMs in the same cluster. For the CMs that cannot be kept in the same cluster due to connectivity problem, they need to initiate adding algorithm to find new attached clusters. Then, the hierarchical levels may be adjusted accordingly.

Figure 5.8: Removing Algorithm

due to the connectivity problem, those CMs need to find new attached clusters. The searching and attaching process needs $O(|V_j||S(\alpha_i, \beta_i)|)$, $j \in S(\alpha_i, \beta_i)$, operations in the worst case because each node $j$ in $S(\alpha_i, \beta_i)$ may need to search their neighbor nodes, i.e., $V_j$, for attaching. After that, in order to maintain the hierarchical clusters, the hierarchical levels may be adjusted for all clusters that are under the lower level of the failed cluster. The searching and updating process is the same as that in case 2 for one node in $S(\alpha_i, \beta_i)$. The difference is that all of the nodes in $S(\alpha_i, \beta_i)$ may need to run
A. Connectivity exists between two clusters even though the failing node is a CM and CN
B. The cluster disconnected due to the failing node searches the new attached cluster

Figure 5.9: Node Removing Algorithm in Case 2.

A. Mesh nodes in the cluster can be reconstructed into a new cluster after negotiation. B. Connected nodes in the cluster can be splitted into different clusters without changing the original hierarchical connectivity with other clusters. C. Disconnected nodes in the cluster can be added into the hierarchical clusters with the change of hierarchical connectivities among the clusters at the lower levels

Figure 5.10: Node Removing Algorithm in Case 3.

this searching and updating process. Thus, the complexity of the removing algorithm in case 3 is $O(\beta_{max} \cdot |S|_{max}^2)$ in the worst case. Several results of removing algorithm in case 3 can be shown in Fig. 5.10.

Now, we have obtained the adaptive algorithms to construct hierarchical clusters based on the service availability. Since the security in the construction is also needed, we introduce flexible authentication protocols next, which are entangled with the construction algorithms to guarantee the security.
5.3 Flexible Authentication Protocols on the Hierarchical Clusters

When we need to authenticate a node on the hierarchical clusters, three scenarios may happen [82]:

- **Case 1**: when a node joins a network for the first time, a strong authentication from the node to its home authentication server should be implemented.
- **Case 2**: when a node joins a cluster, a local authentication for the node is required to reduce the overhead and power consumption of authentication.
- **Case 3**: when a node in a cluster wants to communicate with another node, a session key should be generated for this communication session.

Although the solutions for these scenarios have been proposed in some papers [10, 82], they are not sufficient to resolve the problems due to two reasons. First, they are not involved in the construction of hierarchical clusters, which leaves a breach of security that may further compromise the whole network. Second, they all employ certificate and public/private key based algorithms, which consumes more power than secret key based authentication.

Thus, we propose a solution based on Fig. 5.1. The solution is involved with construction of hierarchical clusters and employs challenge/response authentication with efficient distribution of local security associations of nodes, which is an extension of our previous work.

5.3.1 Management of Security Associations on Hierarchical Clusters

The challenge/response authentication is a secret-key based authentication mechanism implemented with security associations. A security association is a one-way trust relationship between communicators that affords security service on the traffic with parameters of encryption/decryption method, shared key and lifetime. When a challenge/response authentication is used, a node that wants to be authenticated needs to send a request to its authenticator for a challenge value, a random number. After the authenticator replies a challenge value, the node encrypts the challenge value and
replies the result, so-called response, to the authenticator. After verification, the node can be authenticated and authorized.

However, since the nodes in MANet can randomly move from time to time, the security associations may not be available anywhere for the authenticators to verify the nodes. Thus, an intelligent management of security associations is necessary for secret-key based authentication in MANet, which should be highly scalable in MANet. Since hierarchical architecture has been proved to be efficient and scalable [79], managing the security associations on hierarchical clusters becomes a promising way for secret-key based authentication.

In proposed scheme, the management of security associations on hierarchical clusters meets the following requirements after their initial authentication in case 1 to their home authentication servers:

- All the MNs share security associations with the local authentication server that is connected with the GN.
- A CH is trusted by the upper CH, and it also have the security associations with the lower CHs controlled by itself.
- A CH shares security associations with the cluster members controlled by itself.

An example that meets these requirements is shown in Fig. 5.1. A gateway node is connected to a local authentication server, which is responsible for the authentication of MNs in the MANet. The local authentication server stores all the security associations shared with the MNs in the ad hoc network, i.e., \( N_i^{h/c}(\alpha, \beta) \) \((i \in [1, 15], \alpha \in [1, 3]), \) and \( \beta \in [1, 4] \). Given cluster \( S(1, 1) \) is the upper cluster of \( S(2, 1) \) and \( S(3, 1) \), cluster head \( N_{1}^{h}(1, 1) \) needs to store the security associations of \( N_{10}^{h}(2, 1) \) and \( N_{3}^{h}(3, 1) \), which are the CHs of \( S(2, 1) \) and \( S(3, 1) \), respectively. In addition, CH \( N_{11}^{h}(1, 1) \) needs to store the security association with GN, \( N_{2}^{h}(1, 1), N_{6}^{h}(1, 1) \), and \( N_{14}^{h}(1, 1) \), because GN is the upper CH of \( N_{1}^{h}(1, 1) \) and the rest of the nodes are the cluster members of \( N_{1}^{h}(1, 1) \). Similar with \( N_{12}^{h}(1, 1), N_{6}^{h}(1, 2) N_{10}^{h}(2, 1) \), and \( N_{15}^{h}(3, 1) \) are all cluster heads in their clusters, therefore they share the security associations with their cluster members and the CHs that control them and are controlled by them.

However, since each node only shares a security association with its home authentication server initially, how to manage the local security associations on the
hierarchical clusters becomes a critical problem. In order to solve this problem, we embed the management of security associations into the authentication in different cases, which are described next.

5.3.2 Authentication for Newly Joining Nodes

In our system model, an MN shares a security association with its home authentication server. Thus, when a node joins in a MANet for the first time, we require that the node be authenticated to its home authentication server through its CH and GN. After that, two local security associations are established for the node. One is at the local authentication server associated with the GN; the other is either at the CH of the node if the node becomes a cluster member, or at the upper cluster head of the node if the node becomes a CH. The signaling procedure of the authentication in case 1 is shown in Fig. 5.11.

![Diagram](Figure 5.11: Authentication in Case 1.)

In this figure, a connected CH (CCH) is a CH that controls the cluster, which the joining node contact for initial authentication. An intermediate CH (ICH) is the CH that takes charge of sending the authentication request between the CCH and the GN. Then, a process of the authentication in case 1 can be described as follows.

When a CCH receives an authentication request from a joining node, the CCH generates and replies a challenge value, \( V_C \), to the MN. The MN encrypts \( V_C \) into a response value, \( V_R^{(1)} \), with the security association shared with its home authentication server as:

\[
V_R^{(1)} = \{ID, V_C\}K_0,
\]  

(5.26)
where ID is the identification number of the MN, $K_0$ is the key in the security association shared between the MN and its home authentication server. Then, the MN replies $V_R^{(1)}$ back to the CCH.

Since the CCH is lack of credentials such as the key to decrypt $V_R^{(1)}$, the CCH relays the $V_C$ and $V_R^{(1)}$ to the local authentication server through the ICHs and the gateway node. The message, $MSG_{CCH\rightarrow L}$, transmitted between CCH, ICHs and the gateway node is defined as:

$$MSG_{CCH\rightarrow L} \triangleq \{ID, ID_{CH}, V_R^{(1)}, RAN_{CH\rightarrow L}\}^{K_{CH\rightarrow L}}, \quad (5.27)$$

where ID is the identification number of the MN, and $ID_{CH}$ is the CH of the MN. If the MN becomes a cluster member, $ID_{CH}$ is its CH. If the MN becomes a CH, $ID_{CH}$ would be its own ID. $K_{CH\rightarrow L}$ is the key in the security association between the CH and the local authentication server, and $RAN_{CH\rightarrow L}$ is a random number generated to avoid replay attack between the CH and the local authentication server.

When the local authentication receives $MSG_{CCH\rightarrow L}$, it obtains $V_R^{(1)}$ and transmits a message $MSG_{L\rightarrow H}$ to the home authentication server, which is defined as:

$$MSG_{L\rightarrow H} \triangleq \{ID, V_R^{(1)}, RAN_{L\rightarrow H}\}^{K_{L\rightarrow H}}, \quad (5.28)$$

where $K_{L\rightarrow H}$ is the key in the security association between the local authentication server and the home authentication server, and $RAN_{L\rightarrow H}$ is a random number generated to avoid replay attack between the local and home authentication servers.

Then, the home authentication server can verify the MN by decrypting $V_R^{(1)}$ with the key $K_0$ in the security association shared with the MN. After that, the home authentication server generates a key $K_1$ and sends it back to the local authentication server to establish a local security association for the MN. The generation of $K_1$ can be shown as:

$$K_1 = HMAC - SSH2(K_0, \{RAN_1\|ID\}), \quad (5.29)$$

where $HMAC - SSH2$ is a one way hash function, $K_0$ is the key in the security association shared between the MN and the home authentication server, $RAN_1$ is a random number, $ID$ is the identification number of the mobile user, and the symbol $\|$ means the two values are linked together. Then, a message $MSG_{H\rightarrow L}$ is generated as:

$$MSG_{H\rightarrow L} \triangleq \{ID, K_1, RAN_{L\rightarrow H}, \{K_1\}_K_0\}^{K_{L\rightarrow H}}. \quad (5.30)$$
When the message $MSG_{H\rightarrow L}$ arrives at the local authentication server through the Internet, the local authentication server establishes a local security association, $SA_{L-n}$, for the MN as:

$$SA_{L-n} \triangleq \{ID, ALG_{L-n}, K_1, LT_1\}, \tag{5.31}$$

where $ID$ is the identification number of the MN, $ID_{CH}$ is the identification of the CH of the MN, $ALG_{L-n}$ is the algorithm chosen for future local authentication, $K_1$ is the key generated by the home authentication server of the MN, and $LT_1$ is the life time of this security association, which is set to infinity until a failure of the node due to power or movement is detected.

If the MN is a CH, i.e., $ID = ID_{CH}$, the local authentication server saves the $ID$ of the node and the $ID$ of the ICH that relays this message as a record. The record is defined as $RD_n$, which is shown as,

$$RD_n \triangleq \{ID, ID_{ICH}, TIME\}, \tag{5.32}$$

where $ID$ is the identification of the authenticated node, $ID_{ICH}$ is the identification of next ICH that relays the message $MSG_{ICH\rightarrow ICH}$, and $TIME$ is a time stamp to create the record.

Then, a message, $MSG_{L\rightarrow ICH}$, is generated and sent to the CCH through ICHs. $MSG_{L\rightarrow ICH}$ is defined as:

$$MSG_{L\rightarrow ICH} \triangleq \{ID, ID_{CH}, RAN_{K_2}, \{K_1\}_K_0, \{K_2, ALG_{L-n}\}_K_1, RAN_{ICH\rightarrow L}_{K_{ICH\rightarrow L}}\}, \tag{5.33}$$

where $K_2$ is the key generated by the local authentication server to establish the second security association for the MN at the CCH of the node, $RAN_{K_2}$ is the random number used to generate $K_2$, $RAN_{ICH\rightarrow L}$ is a random number to avoid replay attack between the local authentication server and the ICH to which it sends message $MSG_{L\rightarrow ICH}$. The generation of $K_2$ in the local authentication server is similar with the generation of $K_1$ at home authentication server:

$$K_2 = HMAC - SSH2(K_1, \{RAN_2||ID\}), \tag{5.34}$$
where $K_1$ is the key generated by and transmitted from the home authentication server of the MN, $RAN_2$ is a random number, and $ID$ is the identification number of the MN.

When the ICH receives the message $MSG_{L-ICH}$, it decrypts the message, replaces $RAN_{ICH-L}$ with a new random number, and encrypts the new message with a new key $K_{ICH-ICH}$ in the security association shared between it and the next ICH to transmit the message. The new message $MSG_{ICH-ICH}$ is defined as:

$$MSG_{ICH-ICH} \doteq \{ID, ID_{CH}, K_2, \{K_1\}_{K_0}, \{K_2, ALG_{L-n}\}_{K_1}, RAN_{ICH-ICH}\}_{K_{ICH-ICH}}$$  \hspace{1cm} (5.35)$$

Then, if $ID = ID_{CH}$, the ICH saves a record $RD_n$ defined as (5.32) for the authenticated node to a table. All of the ICHs perform the same operations until the CCH receives the message $MSG_{ICH-CCH}$, which is defined as:

$$MSG_{ICH-CCH} \doteq \{ID, K_2, \{K_1\}_{K_0}, \{K_2, ALG_{L-n}\}_{K_1}, RAN_{ICH-CCH}\}_{K_{ICH-CCH}}$$  \hspace{1cm} (5.36)$$

where $RAN_{ICH-CCH}$ is a random number to avoid replay attack, and $K_{ICH-CCH}$ is the key in the security association shared between the ICH and the CCH.

When the CCH receives message $MSG_{ICH-CCH}$, it can obtain $K_2$ and generate the second local security association for the MN as:

$$SA_{CH-n} \doteq \{ID, ALG_{CH-n}, K_2, LT_2\}$$  \hspace{1cm} (5.37)$$

where $ID$ is the identification number of the MN, $ALG_{CH-n}$ is the algorithm chosen for future local authentication, $K_2$ is the key generated by the local authentication server, and $LT_2$ is the life time of this security association, which is set to infinity until a failure of the node due to power or movement is detected. Similar with ICHs and the local authentication server, if the MN is a CH, a record, which is defined as (5.32), is saved at the CCH for the node.

Then, a message $MSG_{CH-n}$ defined as follows is generated and sent to the MN:
\[ MSG_{CH\rightarrow n} \triangleq \{K_1\}_{K_0}, \{K_2, ALG_{L-n}\}_{K_1}, \{\text{RAN}_{CH-n}, ALG_{CH-n}\}_{K_2} \]  \tag{5.38}

where \( \text{RAN}_{CH-n} \) is a random number used to avoid replay attack between the CCH and the MN.

When the MN receives message \( MSG_{CH\rightarrow n} \), it decrypts the first part of \( MSG_{CH\rightarrow n} \), and obtain \( K_1 \). Furthermore, \( K_2 \) and \( ALG_{L-n} \) can be obtained by decrypting the second part of \( MSG_{CH\rightarrow n} \). After decrypting the third part of \( MSG_{CH\rightarrow n} \), the MN acquires necessary parameters and sets up two security associations with CCH and the local authentication server, respectively. The authentication in case 1 is finished.

### 5.3.3 Authentication for Nodes Switching Clusters

When a MN wants to change its cluster, the authentication in case 2 is initiated. In this case, since the MN stores a security association shared with the local authentication server, it can ask the new CCH to authenticate it there. The authentication diagram in case 2 is shown in Fig. 5.12.

![Authentication for Nodes Switching Clusters Diagram](image)  

**Figure 5.12: Authentication in Case 2.**

As shown in this figure, when a MN receives a challenge value after it sends an authentication request to the new CCH, the MN encrypts the challenge value into a response value, \( V^{(2)}_R \), with the security association shared with the local authentication server, and then sends it to the new CCH.
server. Here, \( V_R^{(2)} \) is defined as:

\[
V_R^{(2)} \triangleq \{ID, V_C\}_{K_1},
\]

(5.39)

where \( ID \) is the identification number of the MN, \( V_C \) is a random number generated by the new CCH, and \( K_1 \) is the key in the security association shared between the MN and the local authentication server.

After receiving \( V_R^{(2)} \) from the MN, the new CCH must relay \( V_C \) and \( V_R^{(2)} \) to the local authentication server because the new CCH is lack of credentials to verify them. A message \( MSG_{nCCH\rightarrow L} \) is generated as follows:

\[
MSG_{nCCH\rightarrow L} \triangleq \{ID, ID_{CH}, V_R^{(2)}, RAN_{nCCH\rightarrow L}\}_{K_{nCCH\rightarrow L}},
\]

(5.40)

where \( ID \) is the identification number of the MN, \( ID_{CH} \) is the identification of the CH of the node. If the node becomes a cluster member, \( ID_{CH} \) is the CH of the node. If the node becomes CH, \( ID_{CH} \) is the ID of the node. \( K_{nCCH\rightarrow L} \) is the key in the security association between the new CCH and the local authentication server, and \( RAN_{nCCH\rightarrow L} \) is a random number generated to avoid replay attack between the CH and the local authentication server. Then, the message \( MSG_{nCCH\rightarrow L} \) is sent to the local authentication server through ICHs.

When the local authentication server receives the message \( MSG_{nCCH\rightarrow L} \), it decrypts the message and verifies the MN through the existing security association between the node and the server. If the node can be verified and there is a record of the node at the local authentication server, the local authentication server initiates a process to establish new records of the MN at the ICHs. The establishment process is the same as that in case 1, in which we set up records of nodes at ICHs if the node is a CH, and store a local security association in the new CCH for this node. The difference from that in case 1 is the authentication does not need to contact the home authentication server of the MN because the local authentication server has the credential for this node now.

On the other hand, the movement of a node may leave records at the old ICHs and a security association at the old CCH, which increases the redundancy of the records on many nodes. In order to solve this problem, we require that a node periodically broadcasts a beacon to show its existence. If a node moves out of the range
of a CCH, the CCH will detect the disappearance of the node after a period of time, i.e., $T_b$. After that, the CCH sends a message to notify the corresponding ICHs to remove the records of the node. Thus, the redundancy of the records will be removed.

Finally, a new key $K_3$ will be generated at the local authentication server, and a new security association will be distributed and shared between the local authentication server and the new CCH. Some new records will be stored at the new ICHs between the new CCH and the local authentication server.

5.3.4 Authentication for Nodes Initializing Communications

When a MN wants to communicate with another node in this MANet, a session key should be generated for this communication session, thus the authentication in case 3 is initiated. In this case, we call hierarchical state routing protocol (HSR) to forward the authentication message in case 3 from the source node to the destination node [36].

In the HSR, when a node wants to send messages to another node, it checks its routing table first to find whether the route to the destination node exists or not. If the route exists, the source node just relays the messages to the destination node according to the route. If the route does not exist, the source node relays the messages to its CH. The CH will repeat the same process for the transmission. Since a CH has the information for all nodes that are under its control, the messages can reach the destination node finally because the root of the hierarchical clusters, i.e., GN, keeps the information for all nodes in this MANet.

Therefore, when a source node wants to communicate with another node, it depends on the hierarchical clusters to deliver the session key with hop-by-hop security associations. The signaling diagram of the authentication between two nodes, A and B, in case 3 is shown in Fig. 5.13.

In this figure, when MN $A$ wants to communicate with MN $B$, it generates a message, $MSG_{AB}$, and sends the message to its CH if it cannot find the node $B$ in its routing table. The message $MSG_{AB}$ is defined as:

$$MSG_{AB} \triangleq \{ID_{CH_B}, K_{AB}, RAN_{AB}, RAN_{A-CCH}\}^{K_{A-CCH}}, \eqno(5.41)$$

where $K_{AB}$ is the session key generated by MN $A$ for this communication session, $RAN_{AB}$ is a random number used to avoid replay attack between MN $A$ and $B$ for this
key exchange, $RAN_{n-CCH}$ is a random number to avoid replay attack between MN $A$ and its CCH, and $K_{A-CCH}$ is the key in the security association between MN $A$ and its CCH.

When the CCH of MN $A$ receives message $MSG_{AB}$, the CCH searches $ID_{CH_B}$ in its table that records the identifications of the nodes under its control. Then, message $MSG_{AB}$ is decrypted, encrypted and relayed hop by hop to the MN $B$ with HSR protocol.

When the MN $B$ receives the message that contains the key $K_{AB}$ and random number $RAN_{AB}$, MN $B$ replies the value $\{RAN_{AB}\}_{K_{AB}}$ to MN $A$ through the reversed route.

After the MN $A$ receives the confirmation of $RAN_{AB}$, it knows the key $K_{AB}$ is successfully received by the MN $B$. Thus, a session key is distributed and the authentication in case 3 is finished.

### 5.4 Simulation

In this section, we simulate proposed clustering and authentication schemes in a MANet. We introduce the simulation configuration first, then observe the performance variation with simulation time and mobility of MNs in the second and third parts of this section.

#### 5.4.1 Simulation Configuration

In order to reveal the performance of proposed clustering and authentication schemes, we use Glomosim for our simulations [89].
In our simulation, we have thirty nodes uniformly distributed in an area of 1500m × 1500m at the beginning. Then, we apply random way-point mobility model to twenty nine nodes and leave one node’s position fixed, which acts as the GN shown in Fig. 5.1. For MAC protocol, we use 802.11 for the simulation and set the bandwidth as 11Mbits per second.

In the simulation, we compare the results with a flat MANet and a hierarchical MANet. The flat MANet is using ad hoc on-demand distance vector (AODV) routing protocol [68], and the hierarchical MANet is built by utilizing linked-cluster algorithm (LCA) [4] with hierarchical state routing (HSR) protocol working on it [36]. For comparison, we also implement HSR on the hierarchical MANet built with proposed clustering scheme. Thus, when we simulate the flat MANet, we use AODV as the routing protocol; when we simulate the hierarchical MANet, which is built either with LCA or with proposed clustering scheme, we implement HSR for the routing on hierarchical MANet.

We let six nodes to communicate with each other using constant bit rate (CBR) traffics. Each traffic is initiated after the start of the simulation and sends out 512-bit data every one second until the end of the simulation. In order to calculate the energy consumption, we implement a battery in each node that stores energy. Since we assume that the amount of the energy on each node could be different and has a threshold of value, below which the node cannot transmit any data, we let the amount of the energy on each node uniformly distributed within a scope. Because we want to observe the energy exhaustion during the simulation time, we set the minimal value of the uniform distribution of the energy equal to the threshold value of the energy, i.e., 50 Joule. The energy scope for the uniform distribution is 0.8 Joule. When a node transmits a packet, the power for the transmission is 0.66 watt [59]. The energy for the transmission of a packet can be calculated after we obtain the transmission time by using the bandwidth to divide the packet length. On the other hand, the power for a node to receive a packet is 0.395 watt [59]. Thus, the change of the energy can be carried out based on the traffic transmitted by the node. Furthermore, we assume that the energy charging rate for a battery on the node is 0.005 watt, the number of events that will charge the battery is a Poisson distribution with mean value of 10 seconds, and the time to charge the battery is an exponential distribution with mean value of 5 seconds. The rest of the parameters, such as transmission rage and channel condition, come from the default configuration.
In order to evaluate the proposed authentication schemes in the simulation, we implement proposed clustering algorithms and authentication schemes with Glomosim. In the clustering process, we fix the position of node 0 and make it work as GN. Other nodes will try to join in the MANet rooted at the GN at the beginning of simulations. Each node periodically broadcasts a beacon to show its existence. The beacon includes some information such as a node ID and evaluation of node service availability. The nodes in the MANet will utilize these information to determine the roles of joining nodes. Before allowing a node to join in the MANet, the authentication in case 1 is needed. For the authentication in case 1, we require that the GN should store the security information for the other nodes. When a node other than node 0 wants to join, it needs to send authentication request to the GN for verification. After receiving the approval, it can determine its role, which could be included in the broadcasted beacon. The authentication process in case 2 is the same as that in case 1 in our simulation because we do not set up roaming nodes from other networks, i.e., we assume that all of the nodes belong to current MANet. In the authentication case 3, the CHs will take charge of exchanging the credentials for communication nodes because the CHs have the hop-by-hop security associations shared among them.

After we define the simulation framework including the parameters, we evaluate the performance of the MANet in two scenarios. In the first scenario, we observe the change of performance every 30 seconds starting from 30 to 300 seconds, while fixing the mobility models of the nodes. In the second scenario, we observe the change of performance in different mobility models of the nodes, while fixing the simulation time at 200 seconds. We run each scenario thirty two times with different random seeds, and calculate the mean and variance of the collected data.

The parameters that we evaluate to see the performance include packet loss, throughput, end-to-end delay, and system service availability. The packet loss is evaluated with the total number of packets that are lost during the simulation. The throughput is evaluated with the number of bits received at the destination nodes within unit time, i.e., bits per second. The end-to-end delay is measured between communication nodes. In our simulation, we sum up all the end-to-end delay on all the communication nodes, and show its variations. The fourth parameter that we evaluate in this simulation is the energy consumption. Since all of the nodes may take part in relaying messages,
the energy consumption evaluated in our simulation includes all of the nodes’ energy consumption. The last parameter that we want to evaluate in this simulation is the system service availability that is defined in (5.3). For this parameter, we calculate the statistic of the node service availability on all the cluster heads in the MANet, and show the trend of it with the variation of simulation time and mobility models.

5.4.2 Performance Variation with Simulation Time

In this section, we show the performance variation with the change of simulation time.

As shown in Fig. 5.14, the number of lost packets increases with the simulation time. The flat MANet with AODV has the least lost packets of all these three scenarios. This is because in the flat MANet with AODV, a node can always broadcast the routing messages to find the destination node if the destination node is not in the routing table. In hierarchical MANet with HSR routing protocol, the process to update the location of an MN depends on the communication among cluster heads. This updating process is slow when MNs keep moving, and the old routing information on CHs may lead to the wrong delivery of packets. Therefore, the number of lost packets due to the mobility of MNs in the flat MANet with AODV is less than that in hierarchical MANet. On the other hand, if we compare the hierarchical MANet with LCA and proposed schemes, we can find that the number of lost packets in the hierarchical MANet with proposed schemes is far less than that in the hierarchical MANet with LCA clustering scheme. The improvement with proposed schemes is about 50%, comparing to the clustering algorithm of LCA. The improvement is obtained because the proposed schemes are able to select the nodes with optimal service availability to become the backbone of hierarchical MANet, thus improving the robustness of message routing. Since LCA only chooses the node with least ID to become the CH, the routing message may be dropped more often if the CHs are out of energy sooner or move more often than the CHs selected in our proposed schemes.

In Fig. 5.15, we demonstrate the change of throughput with the increase of simulation time. As we can see in this figure, the throughput in flat MANet with AODV is bigger than those in hierarchical MANet with LCA and proposed schemes. The throughput achieved in the flat MANet with AODV is 3.9% more than that in
the hierarchical MANet with proposed schemes. And, the throughput achieved with proposed schemes is 4.6% more than that with LCA clustering algorithm. Since the throughput is calculated based on the data received over a period of time, it can be understood easily that the trends of the throughputs in Fig. 5.15 are consistent with the trends of lost packets as shown in Fig. 5.14.

We investigate the variation of end-to-end delay with the increase of simulation time as shown in Fig. 5.16. In this figure, the end-to-end delay on the flat MANet with AODV is greater than those on the hierarchical MANet. This is because the initialization of AODV needs to broadcast messages to find the routes between communication nodes, while the routing on hierarchical MANet does not need initialization. The communication nodes in hierarchical MANet only relay the data to their cluster heads, and the CHs will take care of the routing based on existing hierarchical architecture. Therefore, the end-to-end delay in hierarchical MANet is less than that in flat MANet. In the flat MANet with AODV, after the initialization process to find the route, the nodes that involve in relaying data keep the route for a period of time. Thus, the end-to-end delay will be stable at some level. If the route is disconnected due to node movement
or energy depletion, some nodes such as the source node or the intermediate node need to broadcast the message to find new routes, which causes more end-to-end delay in communications. On the other hand, the routing in hierarchical MANet does not require the initialization process, thus initial end-to-end delay is small. With the increase of simulation time, some nodes in the route will move or consume up the energy. The clustering algorithm may be initiated to construct the hierarchical architecture locally again, and find the new route for the communication. Thus, the end-to-end delay will increase with the simulation time and be stable at a level. Since our proposed clustering schemes are more complicated than LCA clustering algorithm, the processing time makes the end-to-end delay in proposed schemes longer than that with LCA clustering algorithm. By comparing the end-to-end delay at the simulation time of 270 seconds, we can see that the end-to-end delay on the hierarchical MANet with proposed scheme is 15.2% more than that on the hierarchical MANet with LCA scheme, the end-to-end delay in the flat MANet with AODV is 10.5% more than that in the hierarchical MANet with proposed scheme.

In Fig. 5.17, we display the variation of energy consumption in the simulation.
Since the transmission of data consumes an amount of energy and the traffic is CBR that is transmitted every 5 seconds in our simulation, the energy consumption in the simulation is proportional to the simulation time. The energy consumptions in these three scenarios have slight differences. In the flat MANet with AODV, the routing path will be optimized by choosing the one with least number of hops, while in the hierarchical MANet, the routing paths just follow the hierarchical clusters, which may not be optimized. Thus, the energy consumption in the hierarchical MANet is usually more than that in the flat MANet with AODV. On the other hand, in our proposed schemes, the selection of CHs is based on the service availability, thus reducing the possibility of consuming the energy for re-organization of clusters. Thus, the proposed schemes can achieve more efficient energy consumption compared with LCA clustering algorithm. Since the major energy consumption is focused on the message and data transmission, the differences of energy consumption among these three scenarios are not quite big.

Since our final purpose is to improve the service availability in the hierarchical MANet, we evaluate the network service availability, which is defined in (5.3), in the simulation. In Fig. 5.18, we compare the network service availability for the hierarchical
MANet with proposed schemes and LCA clustering algorithm. As we can see, the service availability of the CHs with proposed algorithms is always more than that with LCA. The improvement is about 17.5% at the simulation time of 270 seconds. This benefit of proposed scheme comes from the selection of nodes with optimal node service availability, instead of choosing the nodes with least node identification like LCA. In addition, the system service availability in proposed scheme decrease slower than than in LCA. This is because the LCA always chooses the nodes with least node identification as the CHs, which accelerates the energy consumption of these nodes because the CHs need to consume more energy to take care of the routing and communications for the cluster members.

5.4.3 Performance Variation with Mobility

The performance of communications is also affected by the mobility of mobile users, as we can realize intuitively. In this section, we illustrate the same system metrics as shown in previous section at different speeds.
In Fig. 5.19, we demonstrate the relationship between the number of lost packets and the average speed of MNs. As we can see for all of three scenarios, the number of lost packets increases with the speed. For the AODV in flat network, since each time the source node finds a route dynamically before the communication, the increased speed has less effects on the lost packets than the HSR on hierarchical clusters. In hierarchical clusters, the routes are established with the construction of the architecture. When a node moves, the architecture needs to be updated immediately for correct routing information. Because of the reconstruction delay, the correct routing information may not be obtained in time during the communication. Thus, the HSR on hierarchical clusters will cause more packet loss than the AODV in flat network. On the other hand, proposed clustering algorithms select the nodes with more service availability to become the CHs. Since the CHs take more responsibility of routing in HSR, more frequent changes of CHs in LCA due to node movements cause more packet loss than the proposed clustering algorithms. Therefore, the HSR with LCA clustering algorithm loses the most number of packets as the speed increases. The HSR with proposed clustering algorithms loses the second most number of packets. And, the AODV in the flat network loses least
number of packets. At the speed of 9 m/s, the numbers of lost packets for AODV in the flat network, HSR on proposed clustering architecture, and HSR on LCA clustering architecture are 25, 92, and 122, respectively. Comparing to the LCA clustering algorithm, the improvement of our proposed method is around 24.6% at the speed of 9 m/s.

Figure 5.19: Number of Lost Packets vs. Average Speed of Nodes.

In Fig. 5.20, we show the variations of the throughput in three scenarios. The throughput is measured at the destination nodes and quantified with the ratio of the number of received bits over the simulation time. The trends of the throughput in three scenarios are accordingly decreasing with the increase of the speed of MNs. This phenomenon is consistent with the trends of the packet loss in these three scenarios shown in Fig. 5.19. Since the packet loss of the AODV in the flat network is the least, the AODV in the flat network has the biggest throughput. Accordingly, the HSR with proposed clustering architecture has the second biggest throughput in the communication, and the HSR with LCA clustering architecture has the least throughput in these three scenarios.

We reveal the relationship between the end-to-end delay and the average speed of nodes in the simulation in Fig. 5.21. The end-to-end delay increases with the average
speed of MNs, as shown in Fig. 5.21. For all of the three scenarios, we can understand that any movement of the MNs may result in the re-establishment of the routes. In AODV on the flat network, this may be initiated by the source node. In HSR on the hierarchical network, the retransmission may be conducted after the reconstruction of the hierarchical clusters. Since the increased speed of the MNs enhances the probability of re-establishment of routes, the average end-to-end delay will increase as the average speed of MNs. On the other hand, AODV on the flat network needs to broadcast routing request messages on-demand, which increases the delay to transmit data. For HSR on hierarchical architecture, since the routing information is embedded in the construction of the hierarchical architecture, the source node does not need to look for the route for transmission of data. Thus, the end-to-end delay with HSR on hierarchical architecture is far less than that with AODV on the flat network. For the proposed clustering algorithm, since its construction is more complex than LCA algorithm due to the evaluation of the service availability, proposed algorithm needs more time to process the transmission of the data. Thus, the end-to-end delay with HSR on proposed clustering architecture is more than that with HSR on LCA clustering architecture. At the speed of 9 m/s, the end-to-end delays with HSR on proposed clustering architecture

Figure 5.20: Throughput vs. Average Speed of Nodes.
and LCA clustering architecture are 0.017 seconds and 0.021 seconds, respectively, which are 52.8% and 41.7%, respectively, less than the end-to-end delay with AODV on the flat network.

![End-to-End Delay vs. Mobility Model](image)

**Figure 5.21: End-to-End Delay vs. Average Speed of Nodes.**

In Fig. 5.22, we display the variation of energy consumption in different speed of MNs in three scenarios. For all of the three scenarios, we can see that there is a convex in Fig. 5.22. At first, the energy consumption increases with the increase of the speed of MNs. This is because the increased speed causes more packet loss, and the retransmission of the packets from the source nodes will result in more energy consumption. On the other hand, since the notifications of packet loss also depend on the reversed route in the communication, the notification may not reach the source node for retransmission of data as the speed increases. In this case, the lost packets will not lead to the retransmission of the data. Instead, the source node just thinks that the sent packets arrive at the destination node. As the ratio of this type of packets increases, the energy consumption will decrease with the increase of the node speed. Therefore, we can find a convex appearing in these three scenarios. As for the difference between these scenarios, because the energy consumption is mostly related to the packet loss and retransmission, it is easy to understand that the AODV on the flat network has the least
energy consumption of these scenarios due to the least packet loss. Accordingly, the HSR on proposed clustering architecture has the second biggest energy consumption, and the HSR on LCA clustering architecture has the most energy consumption at the fixed average speed of MNs. The difference of the energy consumption between these three scenarios is varied. At the speed of 9 m/s, the energy consumption of proposed clustering architecture is 3.7% more than that of AODV on the flat network, and 1.7% less than that of HSR on LCA clustering architecture.

![Energy Consumption vs. Average Speed](image)

Figure 5.22: Energy Consumption vs. Average Speed of Nodes.

In hierarchical architecture, the network service availability should be considered among the MNs especially for the cluster heads, since they take more responsibility for communications. In Fig. 5.23, we evaluate the network service availability in different average speed of MNs. As we can see, the service availability with the HSR on proposed clustering architecture is much bigger than that with the HSR on LCA clustering architecture. The reason is that proposed method always selects the nodes with maximal service availability in a local scope to become the CHs, while LCA only selects the nodes with least node ID to become the CHs. At the speed of 9 m/s, the network service availability using proposed clustering algorithm is around 21.5% more than that using LCA clustering algorithm.
5.5 Conclusions

In this chapter, we propose design and analysis of authentication mechanisms in multi-hop wireless networks, especially in mobile ad hoc network (MANet). In order to improve the service availability and security in MANet, we propose a set of methods, which include robust clustering algorithms based on the evaluation of the node service availability, and flexible authentication protocols on the hierarchical clusters. In the robust clustering algorithms, the construction process considers the quantitative evaluation of service availability with the models of energy and mobility. By handling the joining and leaving cases and the failures caused by these cases, the clustering algorithms are able to improve the robustness of the hierarchical architecture. In addition, in order to secure the clustering process as well as the communications between nodes, the flexible authentication protocols provide secure authentication protocols in three scenarios. These authentication protocols are fully entangled with the construction of hierarchical clusters, and prevent the MANet and communications from unauthorized access successfully. Comparing to the clustering algorithm in the hierarchical clusters, the proposed clustering algorithms and authentication protocols improve the performance of MANet in various scenarios in different communication time and average speed of

Figure 5.23: Network Service Availability vs. Average Speed of Nodes.
group of mobile nodes. Although the proposed clustering algorithms and authentication protocols cannot beat AODV on the flat MANet in some parameters, such as packet loss and throughput, it does exceed the performance of original hierarchical clustering algorithms. Moreover, proposed clustering algorithms and authentication protocols provide security functions to protect the construction of hierarchical clusters and the communications, which are not addressed in either original hierarchical clustering algorithms, or AODV on the flat MANet. Therefore, by considering the energy consumption, mobility, and security of construction and communications in MANet, proposed approaches enhance the service availability and security in MANet significantly, and pave the way for future improvement of service availability in MANet with quantitative analysis of it.
Chapter 6

Conclusions and Future Directions

As the development of wireless communication, the security mechanisms are designed to protect the information secrecy, while imposing more overheads to the data transmission. As one of the most widely used security mechanisms, authentication is used to identify mobile nodes (MNs), prevent unauthorized usage, and negotiate credentials with heavy overhead. Although authentication mechanisms help protect the communication, they induce more overheads, such as encryption/decryption load and long delay, in dynamic network environments. The overheads of authentication have more side-effects on the communication in wireless networks due to the stricter requirements of scalability and security caused by the mobility and open mediums, respectively. Existing solutions present careful protocol designs to reduce the burdens caused by the authentication, whereas they propose little quantitative analysis, flexible protocol design, and optimized architecture implementation on the authentication that are adaptive to the quality of service (QoS) up to date.
6.1 Summary of Thesis

In order to facilitate the communication in authentication, we analyze and design authentication mechanisms in single- and multi-hop wireless networks in this thesis.

6.1.1 Design and Analysis of Authentication in Single-Hop Wireless Networks

The design and analysis of authentication in single-hop wireless networks focus on the scenario where inter-domain roaming happens. The research work in this thesis first analyzes the impact of authentication on the security and QoS, in terms of delay and call dropping probability, simultaneously. Since hierarchical authentication architecture (HAA) is the most widely used in wireless networks, several enhanced authentication protocols are proposed on HAA for net-to-net and mobile-to-net authentication to reduce the authentication delay and cost. Furthermore, considering that the network manageability and security are not sufficient in HAA, we develop a new architecture, which is composed of licensed authentication centers and intelligent control schemes, to reduce the authentication latency, improve network scalability, and enhance the network security in terms of reducing the number of SAs when inter-domain roaming happens.

6.1.2 Design and Analysis of Authentication in Multi-Hop Wireless Networks

When we analyze and design the authentication mechanisms in multi-hop wireless networks, we focus on reliable clustering algorithms to improve the service availability and security, which builds a robust seedbed to implement authentication protocols for secure communications. The reliable clustering algorithms utilize the quantitative evaluation of service availability derived from the energy consumption and mobility of nodes, thus is able to reduce the communication failure effectively. By incorporating the authentication protocols on the clusters in different communication stages, such as construction of hierarchical clusters and initialization of communications, the security
of communications can be guaranteed when mobile nodes frequently join and leave the multi-hop wireless networks.

The numerical and simulation results demonstrate that, by considering the impact of authentication on security and QoS simultaneously in wireless networks, the analysis and design of optimized authentication protocols and architectures can be effective to facilitating the communication efficiency, such as delay and call dropping probability, and improving the network scalability as well as security in terms of reducing number of security associations.

6.2 Future Directions

The quantitative analysis and design of authentication with respect to the QoS parameters build a solid ground for future improvements of communications, also create and envision more challenges about the authentication in wireless networks.

6.2.1 Future Research on the Authentication in Single-Hop Wireless Networks

Most of the authentication mechanisms implemented in the single-hop wireless networks do not adapt to the different requirements of security and QoS of the communications. The proposed analysis and design of authentication protocols and architectures demonstrate a full view to adjust the authentication mechanisms according to the varied environments, such as the increasing number of inter-domain roaming users.

The proposed and optimized authentication protocols focus more on the QoS parameters, such as delay, cost, and life time of security associations, which ignores the optimization of security levels. Since the security levels also affect the system performance, the jointly optimization of security and QoS parameters will be better to improve the integrated system performance in the authentication.

In our proposed design and analysis, we focus more on the effect of authentication either to individual mobile users or between the mobile users and systems. The system-to-system authentication might have more significant impacts on the overall security and QoS of many communication sessions. Thus, a systematic authentication
research should be conducted in order to obtain complete optimization effects during the authentication.

6.2.2 Future Research on the Authentication in Multi-Hop Wireless Networks

Due to the infrastructureless organization in multi-hop wireless networks, current research on authentication mechanisms focus more on the efficient key management in terms of reducing the number of distributed keys and light-weight cryptographic load. Although these parameters are all related to the system performance of multi-hop wireless networks, previous solutions do not have quantitative relationship between the security and system performance. Proposed methods of our thesis in multi-hop wireless networks combine the quantitative evaluation of system performance with the security by requiring the nodes with maximal service availability to become cluster heads and manage the security associations for other members, while leaving several challenges for future development.

Our proposed authentication protocols and clustering algorithms are working on a system model with a gateway node serving for all of the other nodes in this wireless network. With the increasing number of mobile nodes, our design may not be very efficient and scalable to a wireless network of very large scale. Thus, a more scalable and dynamic design, which considers the QoS and security simultaneously, should be proposed.

In our proposed authentication design in multi-hop wireless networks, we only consider a very general security level cooperated with the clustering algorithms. However, as we pointed out in the authentication for single-hop wireless networks, the investigation of different security levels should be conducted to obtain optimized design. In our case for the authentication design in multi-hop wireless networks, a future direction in this field is to optimize the cluster size and hierarchical layers with respect to the security level and different QoS requirements, simultaneously.
Bibliography


