

國立臺灣大學資訊管理學研究所碩士論文

指導教授：林永松 博士

行動 IPv6 中繞送最佳化機制下之  
連結更新認證協定

Authentication Protocols for Binding Update  
in Route Optimization of Mobile IPv6


研究生： 余俊達 撰

中華民國九十三年七月



行動 IPv6 中繞送最佳化機制下之  
連結更新認證協定

Authentication Protocols for Binding Update  
in Route Optimization of Mobile IPv6



本論文係提交國立台灣大學  
資訊管理學研究所作為完成碩士  
學位所需條件之一部份

研究生： 余俊達 撰

中華民國九十三年七月



僅以此文獻給我的親人～





# 謝 詞

這篇論文的完成是我求學階段的一個段落，在這求學的期間一直給予我教誨、支持和幫助的是為我付出最多的父母。感謝我的父親余崑龍先生和母親張春燕女士在生活上、心理上提供我不虞饋乏的幫助，讓我可以專注於學業、充滿勇氣去面對困難。其次是我的哥哥俊德和弟弟俊緯，你們殷切的關心讓我在完成論文的路上帶來一絲溫暖。

在我後期的求學生涯中，影響我最大的莫過於恩師林永松老師。從大學開始就深受老師的吸引，直到研究所之後，老師所教導的數學定理、邏輯推導和嚴謹的求學態度更讓我受益良多。口試期間，承蒙本系莊裕澤主任和蔡益坤老師對論文的修正與建議，並提供不少寶貴的想法，使得本論文得以更臻完善。感謝博士班學長柏皓於論文上的啟蒙，是您引導我進入資訊安全的領域，而且提供相當豐富的資訊和不厭其煩的播空與我討論，不僅如此，也是您讓我有完成這篇論文的勇氣。感謝佩璇在論文初始階段的討論和校稿。

感謝我研究所陪我渡過這兩年的同學們。感謝育先無論在何時都願意伸出手幫助我，讓我不用獨自的處理一切事情。感謝瑩珍、明立、耿宏、大鈞、翔騰、坤威、閔元、柏鈞和榮耀在這兩年學業上的切磋和生活上的分享，讓我的研究生生活充實許多。感謝實驗室學弟妹建宏、孝澤、可愛的琳智、書平和明源在口試期間的幫助和實驗室工作的分擔。

余俊達 謹識

于臺大資訊管理研究所

九十三年七月





# 論文摘要

論文題目：行動 IPv6 中繞送最佳化機制下之連結更新認證協定

作者：余俊達

九十三年七月

指導教授：林永松 博士

行動 IPv6 是在網際網路通訊協定第六版通訊協定上，讓行動裝置在不破壞原有的架構下，更換其連結點卻不影響到其它的網路節點或正在執行的應用程式。為了提供這項服務，行動 IPv6 引入一個新的網路節點：本地代理者 (Home Agent)。本地代理者會記錄行動裝置新取的網路位址，並將送往行動裝置的封包轉送到新的網路位址。然而本地代理者的引進雖然解決了移動的問題，但是新的問題也跟著產生。如果來源端和目的地端相當接近，而本地代理者卻在相當遠的地方，從來源端送給目的地端的封包會先送到本地代理者的所在地再轉送到目的地端，因為來源端和目的地端相當接近，所以這種方法增加很多的傳送延遲。解決這問題的方法就是讓來源端和目的地端直接傳輸，這種方法叫做「繞送最佳化」，但卻引發安全性的問題。

為了解決上述的安全性問題，許多的協定被提出來。在本論文中，我們分析這些協定抵擋攻擊的能力，並考量執行效能。

最後，有三個協定被提出來解決「繞送最佳化」上的安全性問題，其中一個是以現有架構為基礎的協定，另外兩個協定則是以密碼學產生位址 (Cryptographically Generated Addresses) 技術為基礎。以現有架構為基礎的

協定提供最好的安全防護，但是對阻斷式攻擊依然沒有有效的防護。另兩個協定則是將現有利用密碼學產生位址技術協定的安全性漏洞加以填補並改進其執行效率。在本論文的最後，這三個協定的安全性與執行效率將會被完整的分析。

**關鍵詞：**行動 IPv6、鑰匙交換認證、連結更新安全、安全性分析



# THESIS ABSTRACT

GRADUATE INSTITUTE OF INFORMATION MANAGEMENT

NATIONAL TAIWAN UNIVERSITY

NAME : CHUN-TA YU

MONTH/YEAR : JUL, 2004

ADVISER : YEONG-SUNG LIN

## AUTHENTICATION PROTOCOLS FOR BINDING UPDATE IN ROUTE OPTIMIZATION OF MOBILE IPV6

Mobile IPv6 (MIPv6) is a protocol proposed by IETF organization and based on Internet Protocol (IP) version 6 to support mobility. In order to support mobility, MIPv6 uses an additional network node, Home Agent (HA), with a fixed network address. Packets sent to HA first and HA relayed them to the destination. This mechanism introduces another problem: even if the sender and the receiver are close, the sender still has to send packets to the remote HA, which then relay them to the remote receiver. It increases unnecessary routing. This problem is named “Triangle Routing” Problem. IETF proposes “Route Optimization” to solve this problem. The sender sent packets to receivers directly instead of relaying from HA. Although it solves the delay caused by triangle routing problem, it introduces security issue.

Several protocols were proposed to solve security problems in Route Optimization. We list all possible attacks to analyze these protocols. In the protocol analysis, some flaws are found in these protocols. Beside the security issues, performance of the protocol is under consideration.

Three Protocols was proposed in the paper, one is based on existed infrastructure, and the others are based on Cryptographically Generated Addresses (CGA) technology. The infrastructure-based protocol solves most threats in binding update except the Denial of Service (DoS) attack. The other two protocols fix the loophole in other CGA-based protocols and improve the performance. At last, the three protocols are evaluated in security and performance.

**Keyword: Mobile IPv6, Authentication Key-Exchange, Secure Binding Update, Security Analysis**



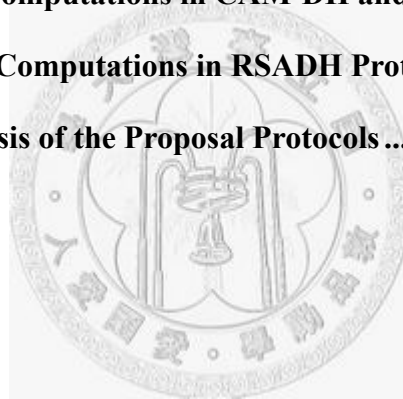
# Table of Contents

謝 詞.....	I
論文摘要.....	III
THESIS ABSTRACT.....	V
Table of Contents .....	VII
Lists of Tables .....	IX
Lists of Figures .....	X
<b>Chapter 1 Introduction.....</b>	<b>1</b>
1.1 Background.....	1
1.1.1 Mobile IPv6 .....	1
1.1.2 Hierarchy of Certificate Authorities.....	8
1.1.3 Cryptographically Generated Addresses.....	10
1.2 Motivation.....	11
<b>Chapter 2 Protocol Survey .....</b>	<b>12</b>
2.1 Threats.....	12
2.2 Protocols Introduction.....	17
2.2.1 Lightweight Protocol .....	18
2.2.2 CGA-based Protocol .....	23
2.2.3 PKI-based Protocol .....	28
2.3 Assessment of Security .....	29
<b>Chapter 3 Solution Approach .....</b>	<b>31</b>
3.1 Problem Description .....	31

3.2 Infrastructure-based Protocol.....	32
3.3 Non-Infrastructure-based Protocol.....	37
3.3.1 RSADH Protocol .....	37
3.3.2 HNK Protocol .....	42
3.4 Binding Update Process .....	45
<b>Chapter 4 Evaluation.....</b>	<b>48</b>
4.1 Assessment of Computational Intensity.....	48
4.1.1 Performance of RSADH Protocol.....	48
4.1.2 Performance of HNK Protocol.....	51
4.2 Assessment of Security .....	51
4.2.1 Security of the FHA-CA Protocol.....	52
4.2.2 Security of the RSADH and HNK Protocol .....	53
4.3 Discussion .....	55
<b>Chapter 5 Conclusion .....</b>	<b>56</b>
5.1 Summary .....	56
5.2 Future Work .....	57
<b>References.....</b>	<b>58</b>

# Lists of Tables

<b>Table 2–1 Basic Assumption of Security in MIPv6 .....</b>	<b>13</b>
<b>Table 2–2 Notation of Cryptography .....</b>	<b>17</b>
<b>Table 2–3 Security Analysis of Previous Protocols.....</b>	<b>29</b>
<b>Table 3–1 Structure of a Question and an Answer.....</b>	<b>39</b>
<b>Table 3–2 Generator <math>g</math> .....</b>	<b>41</b>
<b>Table 4–1 Six Intensive Computations in CAM-DH and SUCV .....</b>	<b>49</b>
<b>Table 4–2 Five Intensive Computations in RSADH Protocol .....</b>	<b>50</b>
<b>Table 4–3 Security Analysis of the Proposal Protocols .....</b>	<b>51</b>



# Lists of Figures

<b>Figure 1–1 Architecture Example of Mobile IP .....</b>	<b>5</b>
<b>Figure 1–2 Triangle Routing Problem.....</b>	<b>7</b>
<b>Figure 1–3 A Hierarchy of Certificate Authorities.....</b>	<b>9</b>
<b>Figure 2–1 Four Types of Active Attacks .....</b>	<b>15</b>
<b>Figure 2–2 RR Protocol .....</b>	<b>19</b>
<b>Figure 2–3 Bake/2 Protocol .....</b>	<b>20</b>
<b>Figure 2–4 S-Bake Protocol.....</b>	<b>22</b>
<b>Figure 2–5 CAM-DH Protocol.....</b>	<b>24</b>
<b>Figure 2–6 SUCV Protocol .....</b>	<b>25</b>
<b>Figure 2–7 ABK Protocol .....</b>	<b>26</b>
<b>Figure 3–1 Trust Relationship of HAs between Different ISPs .....</b>	<b>33</b>
<b>Figure 3–2 Architecture of FHA-CA Protocol.....</b>	<b>34</b>
<b>Figure 3–3 Architecture of RSADH Protocol .....</b>	<b>38</b>
<b>Figure 3–4 Architecture of HNK Protocol.....</b>	<b>43</b>
<b>Figure 3–5 Process of Binding Update .....</b>	<b>46</b>



# Chapter 1 Introduction

## 1.1 Background

We introduce two major parts of knowledge in this section. First, the Mobile IPv6 (MIPv6) architecture is discussed briefly. Afterward, security problems with binding updates (BU) and its solution are included. The second part of the section is threats introduction, these threats can be used to attack binding update mechanism for MIPv6. This information helps us to analyze previous studies in the same region.

### 1.1.1 Mobile IPv6

More and more devices connect to Internet to access resource and communication each other. Most of them are designed to work under the Internet Protocol (IP), thus they need IP address when connecting. A 32-bit IP version 4 (IPv4) address allows approximately four billion IP address for using, however the IP demand will exceed the maximum IP address space in the near future. Thus, IP version 6 (IPv6) was proposed to solve the IP address shortage

problem. The IPv6 specification is defined in RFC 2460 [2].

Without mobility support, packets are routed from a source endpoint to a destination by routers according to their routing tables. Routers collect the nearby Internet topology information to maintain routing tables. When the packet enters the router, the router checks the subnet which packet's destination IP address represents by masking off some of the low-order bits, then retrieves the information from the routing table and relays the packet to next router.

Thus, Mobility is another important issue when various wireless technologies use the IP protocol. When users move with their wireless devices, these devices may change of its attachment point within the Internet topology. Within attachment point change, IP address mostly changes as well. If some applications on these devices were connecting to Internet, the packets send to these applications may be lost because destination IP address was different. The goal of Mobile IP (MIP) is to deal with this problem.

While the designers tried to propose a protocol to support mobility on the Internet, they were faced with two mutually conflicting requirements: (1) when a mobile node changes its IP address, the packets destined to this mobile node must be routed to it correctly, (2) In order to support the Transmission Control Protocol (TCP), the mobile node cannot change its IP address while TCP connecting. If the mobile change its IP address, the connection established by it will be disrupted. MIP was designed to meet these two requirements as primary goal.

Before we describe the detailed working of MIPv6, we present the definition of some of the important terms associated [6].

A **Mobile Node (MN)** is a node that can change its point of attachment from one link to another, while still being reachable via its home address.

A **Correspondent Node (CN)** is a peer node with which a mobile node is communicating. The correspondent node may be either mobile or stationary.

A **Home Link** is the link on which a mobile node's home subnet prefix is defined.

A **Foreign Link** is any link other than the mobile node's home link.

A **Home Address (HoA)** is a unicast routable address assigned to a mobile node, used as the permanent address of the mobile node. This address is within the mobile node's home link. Standard IP routing mechanisms will deliver packets destined for a mobile node's home address to its home link. Mobile nodes can have multiple home addresses, for instance when there are multiple home prefixes on the home link.

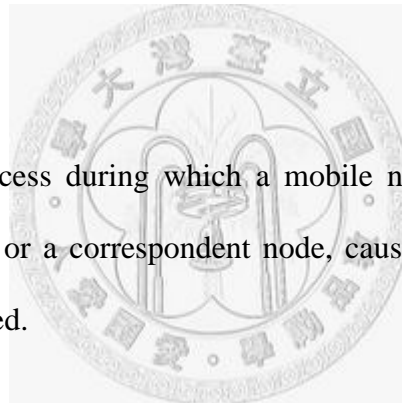
A **Care-of Address (CoA)** is a unicast routable address associated with a mobile node while visiting a foreign link; the subnet prefix of this IP address is a foreign subnet prefix. Among the multiple care-of addresses that a mobile node may have at any given time (e.g., with different subnet prefixes), the one registered with the mobile node's home agent for a given home address is called

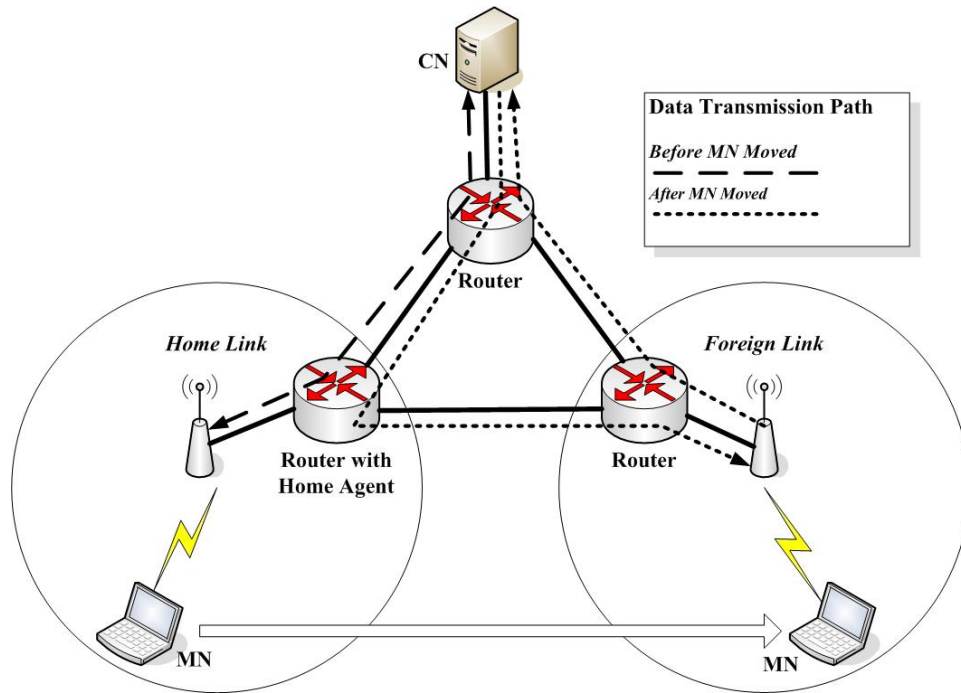
its "primary" care-of address.

A **Home Agent (HA)** is a router on a mobile node's home link with which the mobile node has registered its current care-of address. While the mobile node is away from home, the home agent intercepts packets on the home link destined to the mobile node's home address, encapsulates them, and tunnels them to the mobile node's registered care-of address.

A **Binding** is the association of the home address of a mobile node with a care-of address for that mobile node, along with the remaining lifetime of that association.

A **Registration** is the process during which a mobile node sends a Binding Update to its home agent or a correspondent node, causing a binding for the mobile node to be registered.





**Figure 1-1 Architecture Example of Mobile IP**

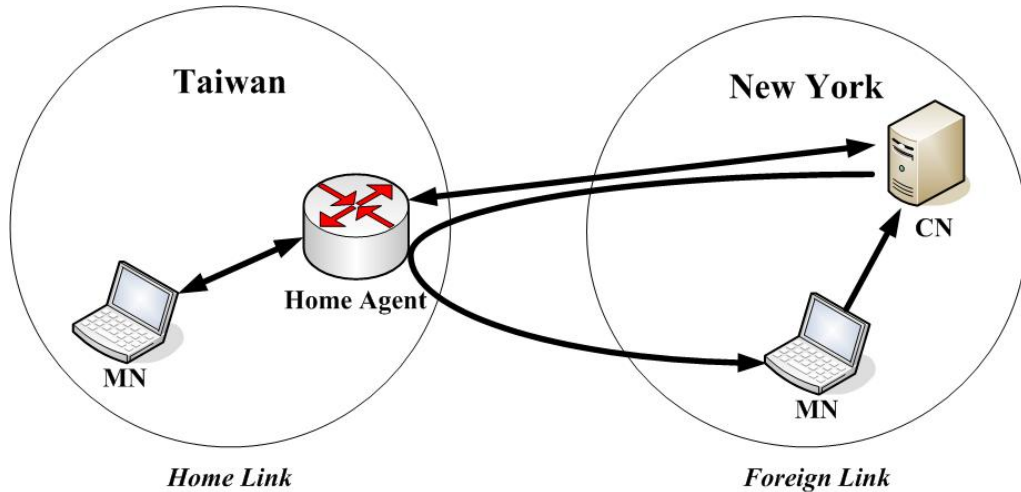
In Figure 1-1, we illustrate an architecture example of MIPv6. It shows how MIPv6 works. In this architecture, the mobile node communicates with the correspondent node, the mobile node is in the home link first. The home agent contains a binding cache which maintains the mobile node's IP. The mobile node has an IP address with home link prefix plus the interface identifier and this IP address is recorded in the binding cache. At this moment, the packets sent from the correspondent node destined to the router whose subnet prefix equals to the mobile node's home link prefix. As definition, the home agent is on the router above discussed. After the home agent received the packets, it relays them to the destination. The data transmission path is shown in figure 1-1 "Before MN Moved".

When the mobile node changes its physical location, it may cause the attachment point change. In this situation, the mobile node moves to a foreign

link and needs to connect to the local network. There are two mechanisms for the mobile node to connect the local network. The mobile node can listen to the network and wait for an agent advertisement (Stateless Address Autoconfiguration [18]) to get a new CoA or may be assigned a new CoA by the DHCP server (Dynamic Host Configuration Protocol [4]). Otherwise, the mobile node can send out an Agent Solicitation actively to negotiate a new CoA from the nearby agent.

After the mobile node got a CoA, it sends a binding update to the home agent, and the home agent will update the binding cache. If this binding update is legitimate, the home agent will send back a binding acknowledgement (BAck) to the mobile node. Once the registration is done, packets sent from the correspondent node will transmit to the home agent first, and the home agent will relay them to the CoA according to the information of binding cache. When the mobile node wants to send packets to the correspondent, it sent them to the correspondent node directly.

Consider the following situation: the mobile node is in the home link, the correspondent node sends packets to the home agent, and then the home agent relays them to the mobile node. If the mobile node moves to a faraway foreign link, the correspondent node has to send packets to the home agent first, and then the home agent relays them through a faraway routing path to the mobile node. Even if the correspond node and the mobile node are close to each other, they still need to send packets through a redundant path. This problem is called “Triangle Routing Problem” as shown in figure 1-2.



**Figure 1–2 Triangle Routing Problem**

To solve the triangle routing problem, the designers proposed a mechanism named “Route Optimization”. After the mobile node moved and registered its new CoA to the home agent by binding update signals, the mobile node send binding updates to the correspondent nodes to inform them of its CoA change. The correspondent node receives the binding update, and then it will check the existence of the mobile node entry. If such entry is existed, the correspondent will check whether the binding update is legitimate. If the binding update is legitimate, the correspondent node will update its binding cache and send a binding acknowledgement to the mobile node, otherwise send a binding error to the mobile node. After the above process was success, the correspondent node knows the CoA of the mobile node, and sends packets to the CoA directly.

Some improvements from Mobile IP version 4 about route optimization was describe following [6]:

- There is no need to deploy special routers as "foreign agents", as in Mobile IPv4. Mobile IPv6 operates in any location without any special

support required from the local router.

- Support for route optimization is a fundamental part of the protocol, rather than a nonstandard set of extensions.
- Mobile IPv6 route optimization can operate securely even without pre-arranged security associations. It is expected that route optimization can be deployed on a global scale between all mobile nodes and correspondent nodes.
- Support is also integrated into Mobile IPv6 for allowing route optimization to coexist efficiently with routers that perform "ingress filtering".

However, the introduction of binding updates and other messages between the mobile node and the correspondent node creates new security problems the need to be addressed. We now discuss these security concerns.

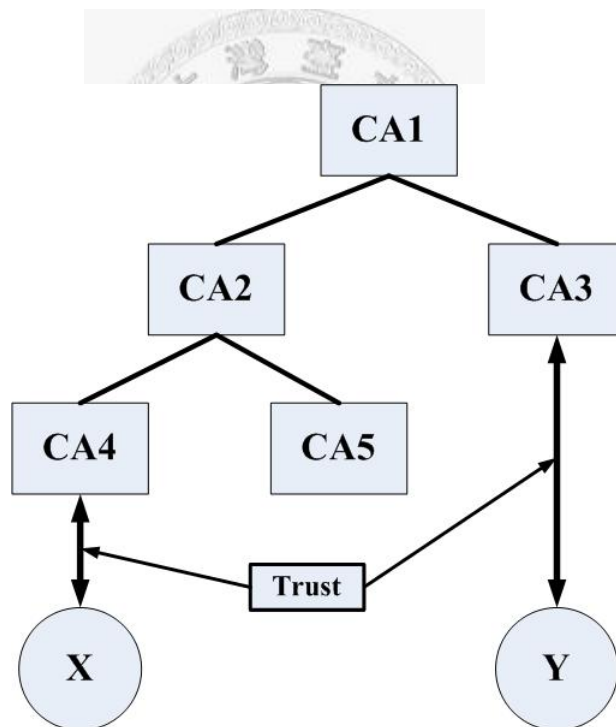
## 1.1.2 Hierarchy of Certificate Authorities

For the identity of the arbitrary node, we need a trusted third party to authenticate it. If the two nodes trust the same third party, they can identify each other by some protocols running within them. However, for arbitrary nodes, it's difficult to trust each other if they don't have a co-trust third party. Thus, a global CA or similar infrastructure (e.g., Public Key Infrastructure (PKI)) is necessary to exist. If every node has its key and store in the global CA, then the CA needs huge memory to store key information. Beside, the key



exchange protocol will run out the resource of the CA, if huge numbers of nodes run protocols with the CA at the same time. In practical, such a CA is very difficult to maintain.

A hierarchical CA architecture was proposed to share load of a global certificate authority. Figure 3-1 shows a hierarchy of CAs. CA2 certifies CA4 and CA5, thus CA2, CA4, and CA5 trust each other. If someone is certificate from CA5, he is certificated by CA1 because of hierarchy relation. In figure 3-1, X trusts CA4 and Y trusts CA3. CA4 Because CA4 and CA3 certificate each other by CA1, X and Y trust each other.



**Figure 1-3 A Hierarchy of Certificate Authorities**

### 1.1.3 Cryptographically Generated Addresses

This technique [10][14] provides an intermediate level of security below strong public-key authentication and above weak authentication [1]. The idea is to form the last 64 bits of the IP address (the interface identifier) by hashing the host's public signature key. Binding Updates can then be signed with this key. A secure one-way hash function makes it difficult for the attacker to come up with a key that matches a given address and to forge signed BUs. The attraction of this technique is that it provides public-key authentication independent of any trusted third parties, PKI, or other global infrastructure.

The main weakness of the scheme is that only 62-64 bits of the IP address can be used for the hash. Thus, an attacker may be able to mount a brute force attack and find a matching signature key by trial and error. Another limitation of the cryptographically generated addresses is that although they prevent the theft of another host's address, they do not stop the attacker from inventing new false addresses with an arbitrary routing prefix. The attacker can generate a public key and a matching IP address in any network and use it to mount bombing. While the public-key protocols (both PKI-based and CGA-based ones) provide a reasonable protection against unauthentic BUs, they are computationally intensive and therefore expose the participants to denial-of-service (DoS) attacks.

## 1.2 Motivation

MIPv6 provides mobility solution based on IPv6 protocol, thus the existed Internet topology just needs only less modified. However there is still some issues in MIPv6 need to be solved. One of them is the authentication problem in routing optimization mode. Several protocols proposed to solve it, some of them need less computing power with more loopholes; some of them need more computing power or extra security infrastructure. Recently, some applications developed with QoS constraints, therefore the latency caused by mobile nodes handoff must be minimal. Complex computation or extra security infrastructure will introduce unnecessary delay when mobile nodes update their new CoA to HA or CN (in routing optimization mode).

Although several protocols was proposed, they need some modified to fix loopholes which may cause known attacks. Light computing protocols (e.g., symmetric key protocol) are easy to retrieve the session key exchanging from communication; intensive computing protocols (e.g., asymmetric key protocol) are harder to break but the adversary still can retrieve the session key if they can change packets between the communicating nodes. We propose one infrastructure-based protocol to solve the authentication problem in routing optimization mode, and two non-infrastructure-based protocols to fix some loopholes existing in others similar protocols and reduce computation.

# Chapter 2 Protocol Survey

Several protocols addressing securing binding update in route optimization mechanism of MIPv6 have been proposed in the recently past years. To the best of our knowledge, we summarize these protocols into three categories. The three categories are classified by consideration under security, performance and architecture. In this chapter, we analysis possible threats in binding update mechanism of MIPv6. Afterward we construct a table to analysis threats in each protocol in Section 2.2.

## 2.1 Threats

Before we analyze any threats of MIPv6's message, we should include following basic assumptions of security in MIPv6 [8] first.

**Table 2–1 Basic Assumption of Security in MIPv6**

**Basic Assumption**

1. The mobile node and the HA have setup a pre-established bidirectional security association (SA) before the mobile node begins to roam and connects to the network from a location that is not its home. This does not imply that the MN has to always boot up at home before roaming onto other networks. The reason for a bidirectional SA is to authenticate the BU as well as the BAck.
2. In most cases there are no existing, established security associations or other security relationships between the mobile node and the correspondent node. In addition neither Certificate Authorities (CA) nor a Public Key Infrastructure (PKI) exist that would enable the establishment of such SAs dynamically. The reason for requiring a SA between the MN and a CN is because the BU sent by the MN to the CN needs to be secured in order to avoid possible threats.

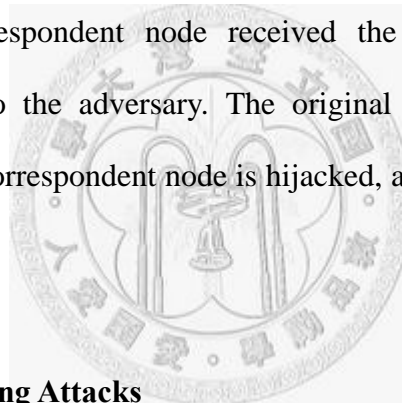
According to above assumption 1, binding updates between the mobile node and the home agent are secure, thus we will focus problem on binding updates between the mobile node and the correspondent nodes.

In general, we classify the attacks on this problem to two primary types: Passive Attack and Active Attack. The passive attack is the adversary can eavesdrop on the link to retrieve the information from packets transmitted between the mobile node and the correspondent node. If there are good protection between the mobile node and the correspondent node, the passive

attack may fail. On the other hand, the binding update is used to redirect traffic from one address to another. If not used carefully, it can have some detrimental effects on the communication between the mobile node and the correspondent node. Binding updates can be used to launch attacks. After analysis, there are four types of active attacks [3][17] which may threaten the binding updates between the mobile node and the correspondent nodes.

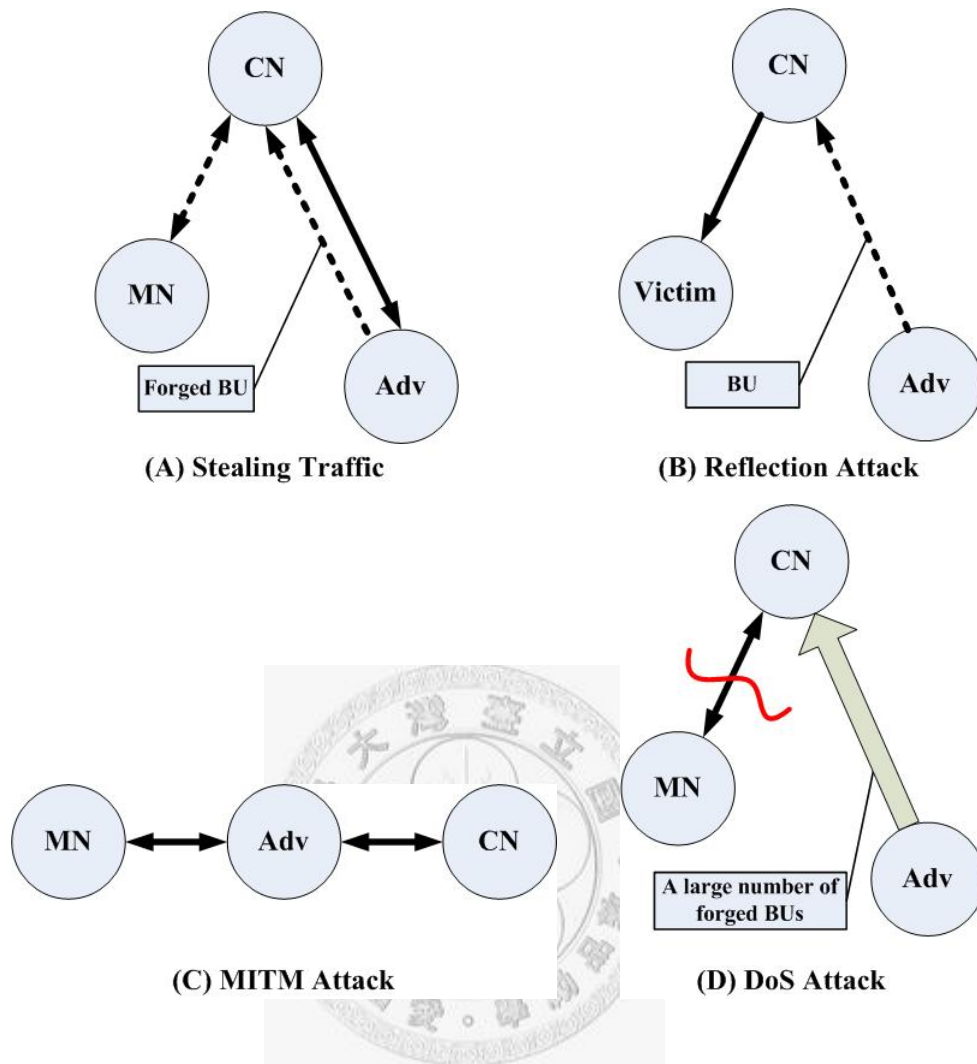
### **1. Stealing Traffic (Session Hijacking)**

If the adversary knows the mobile node's home address, he can impersonate the mobile node and sends a forged binding update to the correspondent node. After the correspondent node received the binding update, he redirects the traffic to the adversary. The original session between the mobile node and the correspondent node is hijacked, as shown in figure 1-3 (A).



### **2. Reflection and Flooding Attacks**

The adversary uses legitimate binding update to redirect the traffic from the correspondent node to the victims, as shown in figure 1-3 (B). For example, the adversary can connect to a video streaming server and send a binding update to this server. This binding update contains the victim's address. Thus, the victim will receive video stream from the video server. The correspondent node sends a lot of traffic to the victims is called flooding or bombing.



**Figure 2-1 Four Types of Active Attacks**

The adversary can generate forged binding updates, and this kind of binding updates source address is the victim's address. The adversary sends them to random correspondent nodes, and then the correspondent nodes will reply binding acknowledgement to the victim. This kind of attack is flooding, too.

### 3. Man in the Middle (MITM) Attacks

The adversary intercepts the binding update sent from the mobile node to the correspondent node. He modifies this binding update and sends it to the

correspondent node. On the other side, it impersonates the original correspondent node to send back a binding acknowledgement to the mobile node. Thus, he can hide between the mobile node and the correspondent node, and eavesdrop or modify the contents of packets as shown in figure 1-3 (C).

#### **4. Denial of Service Attacks**

DoS attacks can be done in several ways, with or without MIPv6. The aim of a DoS attack is to deny service to a legitimate node. The adversary has many ways to achieve his goal. For example, the adversary can send a lot of forged binding update to the victim and then the victim will be busy to process those faked binding update. If some legitimate binding updates come to the victim, they may be ignored because of out of the victim processing power, as shown in figure 1-3 (D). The adversary can use flooding attack to fire a DoS attack.

To avoid above threats, the binding update should be authenticated by the mobile node and correspondent both sides.



## 2.2 Protocols Introduction

In order to explain the protocols easily, we unify the cryptographic notation as following. We will use these notations through the paper.

**Table 2–2 Notation of Cryptography**

<b>Notation</b>	<b>Description</b>
CNA	The address of the correspondent node
$K_X$	A secret key owned by node X.
$K_{BU}$	A session key negotiated from the protocol, used for binding update.
$E(K_X, m)$ or $E(PU_X, m)$	An encryption or decryption algorithm which input is a key and a message. If the key is a secret key, then this algorithm is a symmetric key algorithm. Otherwise, if the key is a public or private key, then this algorithm is an asymmetric key algorithm.
$H(m)$	A one-way hash algorithm which input is a message.
$MAC(K_X, m)$	A message-authentication-code algorithm which input is a secret key and a message.
$PU_X/PR_X$	A public and private key pair owned by node X
$N_{yX}$	A random number produced by node X, y is a number to distinguish the number generated by the same node.
$m n$	Concatenation of two messages m and n.
$g^x$	g is a primitive root which is a well-known system parameter. x is a random number.
$T_0$	Timestamp

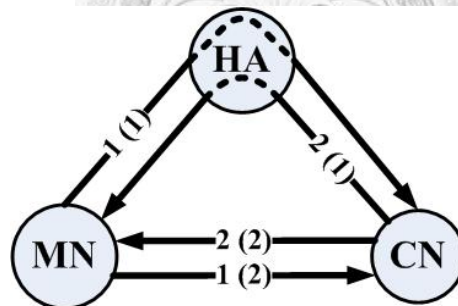
## 2.2.1 Lightweight Protocol

For the reason of optimal performance, the lightweight protocol uses less computational algorithms (e.g., hash function, symmetric key algorithm) to secure the binding updates. Less computational makes mobile node save power, it is important when mobile node is a low-power device.

### 2.2.1.1 Return Routability

#### Protocol Operation

The Return Routability (RR) protocol is proposed in [6] as a default mechanism to secure binding update in MIPv6, as shown in figure 2-1.



Step 1  $1(1) = \{HoA, CNA, N1_{MN}\}$

$1(2) = \{CoA, CNA, N2_{MN}\}$

Step 2  $2(1) = \{CNA, HoA, N1_{MN}, R_H, j\}$

$R_H = MAC(K_{CN}, (HoA | N_{j_{CN}} | 0))$

$2(2) = \{CNA, CoA, N2_{MN}, R_C, i\}$

$R_C = MAC(K_{CN}, (CoA | N_{i_{CN}} | 1))$

$$K_{BU} = H(R_H | R_C)$$

**Figure 2–2 RR Protocol**

First, the mobile node send two messages to the correspondent node, one message go through the home agent, other one go to the destination directly. These two messages are called Home Test Init (HoTI) and Care-of Test Init (CoTI) respectively. After the correspondent node received HoTI and CoTI, he uses his secret key to generate two keygen tokens:  $C_H$  and  $C_C$ , and append them to the Home Test (HoT) and Care-of Test (CoT). While two step processes are success, the session key used for binding update is created by hashing the two keygen tokens.

In the process of protocol, only two symmetric key algorithm computations and a hash function computation are required, thus low computing power needed. However, 1(1) and 2(1) two messages go through the home agent, it causes additional propagation delay. In general situation, propagation delay is larger than symmetric key algorithm computing time. This problem is improved in the Bake/2 protocol.

## 2.2.1.2 Bake/2

### Protocol Operation

The Binding Authentication Key Establishment Protocol version 2 (Bake/2) is proposed in [11] to fix some loopholes caused by RR. In the Bake/2, designers use MAC to keep the integrity of messages. The correspondent node authenticates the messages by his secret key. If the messages were modified, the correspondent node will detect alteration of messages and stop the process. The whole process is shown in figure 2-3.

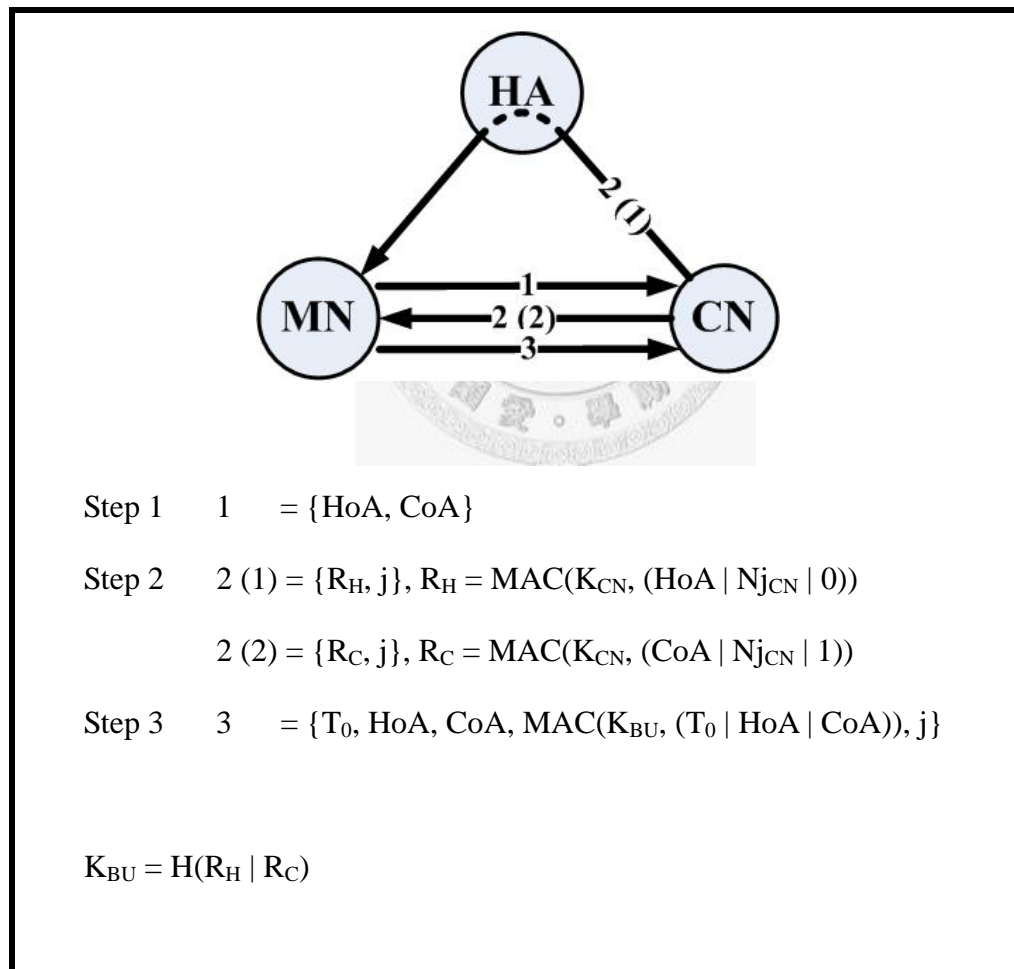


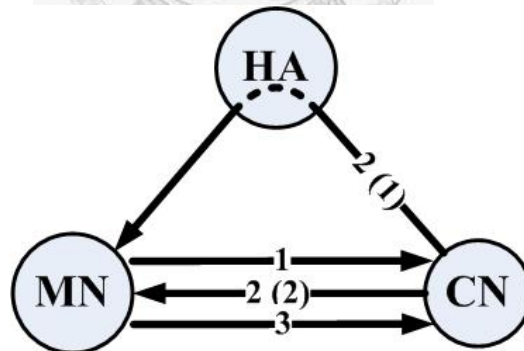
Figure 2–3 Bake/2 Protocol

The correspondent node generates two tokens  $R_H$  and  $R_C$  which are calculated by his secret key. Only the correspondent node knows this secret key, it means the correspondent node can generate  $R_H$  and  $R_C$ . On the other hand, the Bake/2 substitute a message in the RR routing through the home agent for a message sent between the mobile node and the correspondent node.

### 2.2.1.3 S-Bake

#### Protocol Operation

The Simple-Binding Authentication Key Establishment Protocol (S-Bake) is proposed in [15] to reduce computation in the Bake/2 slightly. The S-Bake provides the same security level as the Bake/2 but improves the efficiency. The designers use cookie to replace a computation of MAC. The whole process is shown in figure 2-4.



Step 1    1    = {HoA, CoA}

Step 2    2 (1) =  $\{R_H, j\}$ ,  $R_H = \text{MAC}(K_{CN}, (\text{HoA} \mid N_{j_{CN}} \mid 0))$

          2 (2) =  $\{R_C, j\}$ ,  $R_C = \{\text{Cookie} \mid N_{j_{CN}}\}$

Step 3    3    =  $\{T_0, \text{HoA}, \text{CoA}, \text{MAC}(K_{BU}, (T_0 \mid \text{HoA} \mid \text{CoA})), j\}$

$$K_{BU} = H(R_H | R_C)$$

**Figure 2–4 S-Bake Protocol**

The correspondent node generates two tokens  $R_H$  and  $R_C$  to form the binding update key.  $R_C$  in the S-Bake is different from in the Bake/2; it simply concatenates a cookie and a random number  $N_{j_{CN}}$ . Other steps in this protocol are the same with the Bake/2.



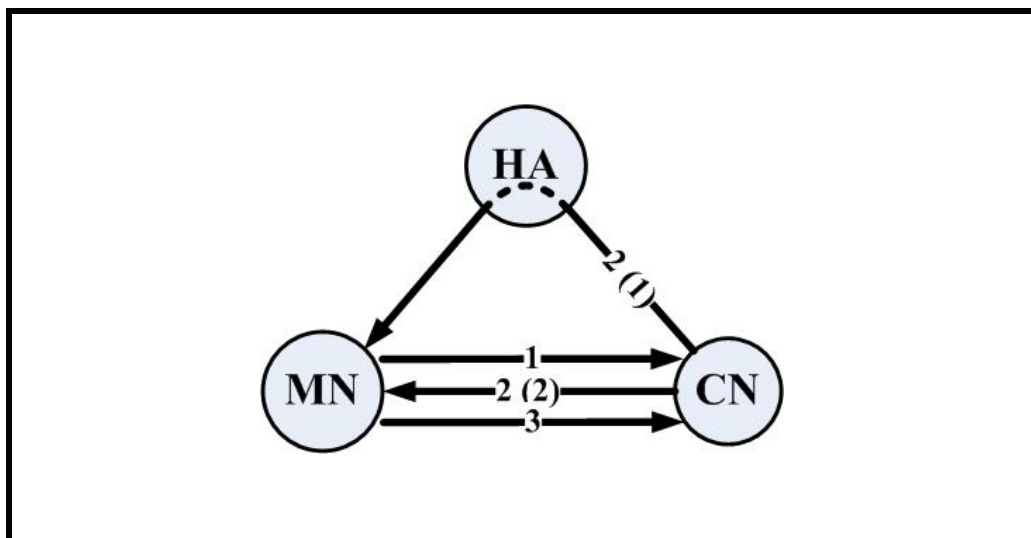
## 2.2.2 CGA-based Protocol

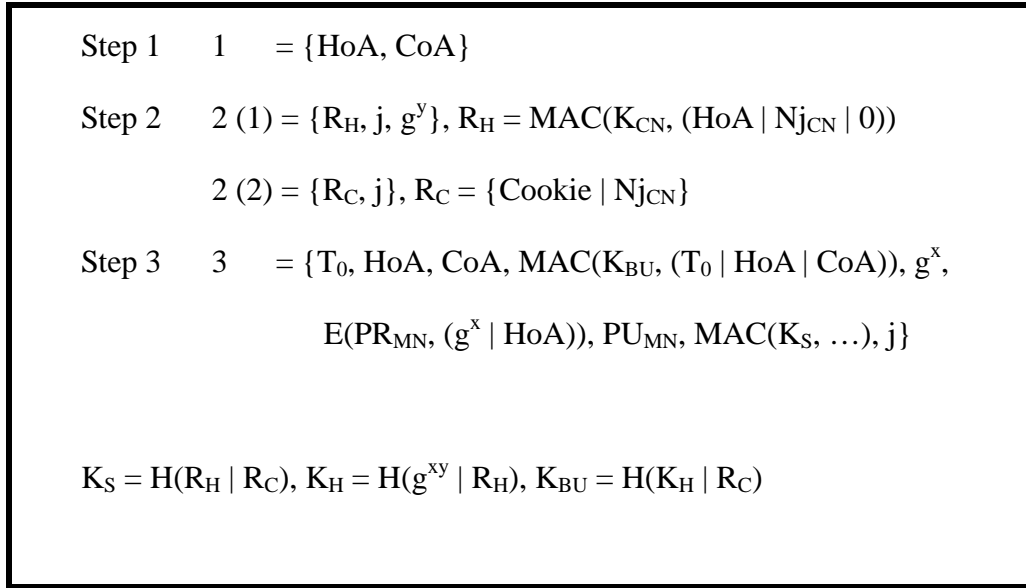
The CGA technology is used to authenticate identity of network node without pre-established infrastructure. For this reason, three protocols were designed by introducing CGA. However, for the fixed address Internet node without supporting mobility, it is hard to find a public key to match the address of this node if this node is not able to change his address. CGA is easy to apply in MIP protocol but hard for IP protocol only. Thus, the protocols applying CGA authenticate the mobile node for the correspondent node, but reverse authenticating depends on whether the correspondent node runs MIP.

### 2.2.2.1 CAM-DH

#### Protocol Operation

CAM-DH [14] protocol combines the BAKE/2 and the CAM (Child-proof Authentication for MIPv6) [12]. CAM-DH uses the same flow of messages in the Bake/2 and a digitally signed DH key exchange in the CAM. The whole process is shown in figure 2-5.





**Figure 2-5 CAM-DH Protocol**

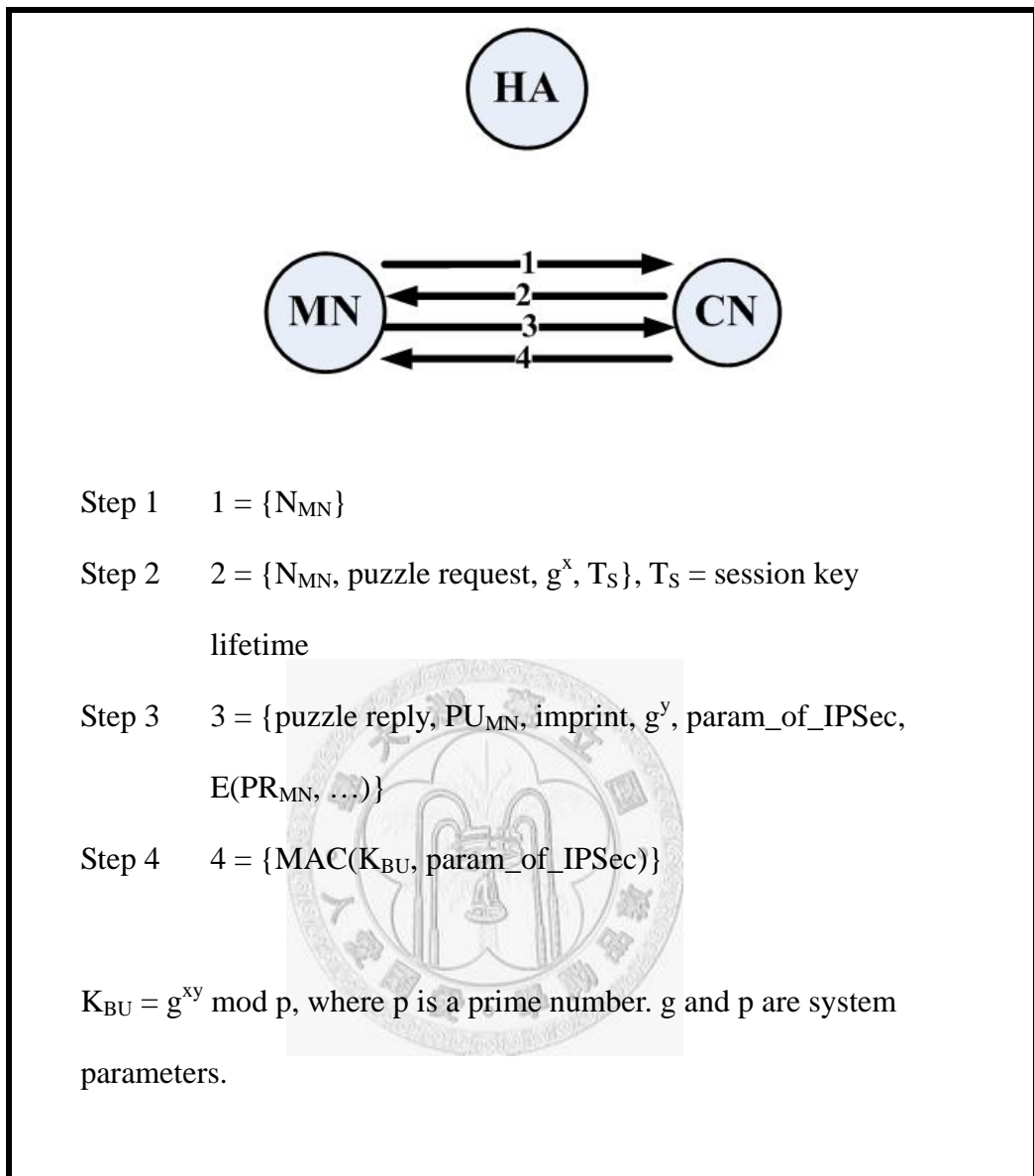
In the CAM-DH protocol, asymmetric key algorithm is used to protect the value of DH ( $g^x$ ), thus much more computation is required. The correspondent can check integrity and identity of  $g^x$  by CGA. Because CAM-DH protocol combines the Bake/2 protocol and the DH key exchange protocol, it has a strong security level than the Bake/2 protocol.

## 2.2.2.2 SUCV

### Protocol Operation

Statistic Uniqueness and Cryptographic Verifiability protocol (SUCV) is proposed in [10] to prevent the impersonation attack. In the SUCV protocol, the interface identity of CoA is the same with the interface identity of HoA. The SUCV protocol introduces the concept of puzzle to prevent DoS attacks. After binding update key is exchanged, the SUCV protocol uses IPsec to protect the binding update messages. The whole process is shown in figure 2-6.





**Figure 2–6 SUCV Protocol**

All messages only communicate between the mobile node and the correspondent node. It takes better performance than message relayed from the home agent. On the other hand, restricted interface identity of CoA makes collision of IP address, but it can be avoid by huge IPv6 address space.

### 2.2.2.3 ABK

#### Protocol Operation

The Address Based Keys protocol (ABK) is proposed in [13] to avoid the need for RR checks on each binding update. The ABK protocol downloads the parameters from the home agent to form the public key of the mobile node, instead of CGA. The partial element of the public key of the mobile node is sent by binding update, and is plaintext in transmission. The whole process is shown in figure 2-7.

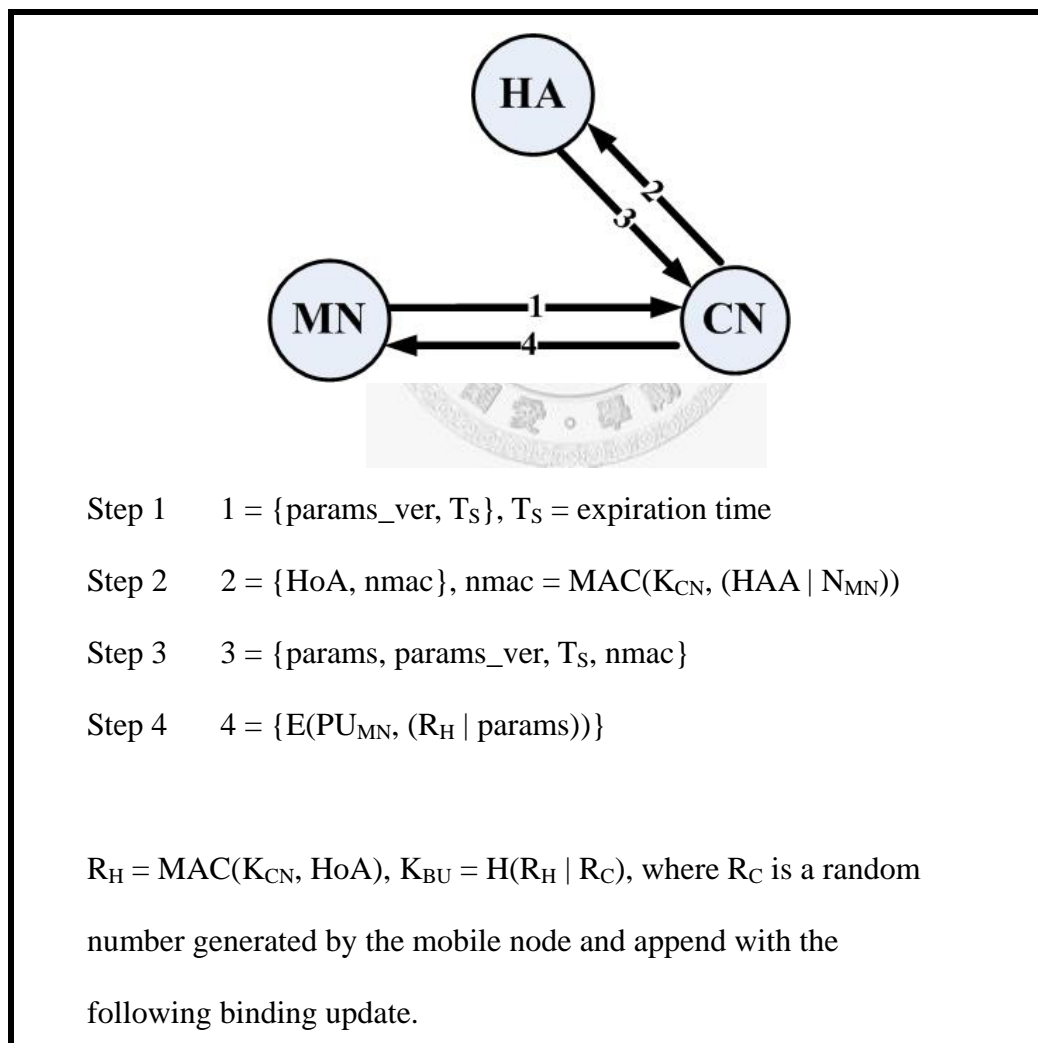


Figure 2-7 ABK Protocol

The designer of the ABK protocol didn't use DH to protect the element of the binding update key. It makes less computation required than the CAM-DH and the SUCV. Moreover, no message is needed to transmit between the mobile node and the home agent, and the transmission time of whole process is reduced.



## 2.2.3 PKI-based Protocol

Several protocols were proposed in this category [5][7][19]. These protocols relay on pre-established public key infrastructure to exchange the binding update key. The pre-established PKI restricts the scalability of the protocols. In CertBU protocol [7], some flawed mechanisms (e.g., SSL) are used if such PKI is not existed.

In the PKI-based protocols, certifications need more transmission time and computation time. If the cross-certification is needed, the complexity of whole process is increased. Under the PKI, the cost of maintenance should be considered. The process of verifying certification also costs much time.

In practice, the mobile node may move its location frequently and thus binding updates are exchanged usually. If every binding update costs too much time to certificate, it will make binding update in route optimization impossible. While a lot of mobile nodes with MIP protocol exist, certifications exchange between these nodes will cause huge load to CAs. The maintenance of CAs will be a big challenge.

## 2.3 Assessment of Security

The summary of security analysis of previous protocols is list in table 2-3. It is useful for designing the protocol of securing binding update. This table will be reused to analyze the proposal protocols in Chapter 4.

**Table 2–3 Security Analysis of Previous Protocols**

<b>Security Analysis</b>					
/	<i>Passive Attack</i>	<i>Active Attacks</i>			
<i>Protocol</i>	Eavesdrop Attack	Stealing Traffic Attack	Reflection Attack	MITM Attack	DoS Attack
RR	√	√	√	√	√
Bake/2	√	√	√	√	√
S-Bake	√	√	√	√	√
CAM-DH	X	X	√ <sub>(1)</sub>	√	X
SUCV	X	X	X	√ <sub>(2)</sub>	X <sub>(3)</sub>
ABK	√ <sub>(4)</sub>	X	X	√	X
PKI-Based	X	X	X	X	√
<p>√: It is not hard for the adversary to use this attack on the protocol</p> <p>X: The protocol can defend against this threat well</p>					

Some special issues in the protocols are discussed. These issues affect the development of the proposal protocols.

1. In the CAM-DH protocol, when the correspondent node received the message from the mobile node, he sends one messages to the home agent and the other to the mobile node. Any adversary can launch a forged binding update to arbitrary correspondent node and let it send double messages to the victims. It is a serious flooding problem from the RR protocol.
2. In the SUCV protocol, the messages are transmitted only between the mobile node and the correspondent node. The correspondent doesn't send a message to the home agent to verify the identity of the home agent. It make the adversary impersonate arbitrary home agent easily and the MITM attack can be also launched easily.
3. The SUCV protocol introduces a puzzle to defend against the DoS attack. The puzzle consumes some power of the mobile node. It makes the attacker must consume more resource than the victim. However, the format of the puzzle is not well defined in [10].
4. The ABK protocol uses two nonces to form the binding update key and the one of the nonces is protected by asymmetric key algorithm. But these nonces are plaintext in transmission; it makes the adversary eavesdrop the nonces easily. When the adversary knows these nonces, he can calculate the binding update key himself.

# Chapter 3 Solution Approach

## 3.1 Problem Description

In this chapter, we focus on the securing binding update transmitting between the mobile node and the correspondent node. There are two directions for us to research: 1. remove the threats and no global central infrastructure needed, 2. if the MITM attack is unavoidable, reduce the complexity of computation and remove other threats which are loopholes of the protocols discussed in chapter 2. In the solution 1, we propose infrastructure-based protocol to achieve these goals. This protocol doesn't rely on global CA or PKI, and it uses the similar idea about hierarchy of CAs. In the solution 2, we propose two non-infrastructure-based protocols to achieve these goals. These two protocols are designed to consume less computation than the previous protocols used CGA. In the both solutions, CGA is used to authenticate the mobile node for the correspondent node. But in the solution 1, the mobile node can authenticate the correspondent node by infrastructure.

## 3.2 Infrastructure-based Protocol

In the previous introduction, a global CA is difficult to maintain. The proposal protocol can't rely on a global CA to authenticate the mobile node and the correspondent node. We take the similar idea of hierarchy of CAs and basic assumption 1 to formulate the protocol. The Friend Home Agent-Certificate Authorities (FHA-CA) protocol is described as following.

### Design Goals

1. Follow the basic assumption as shown in table 2-1.
2. Remove the threats discussed in Section 2.1.
3. Use the MIPv6 architecture efficiently.

### Protocol Operation

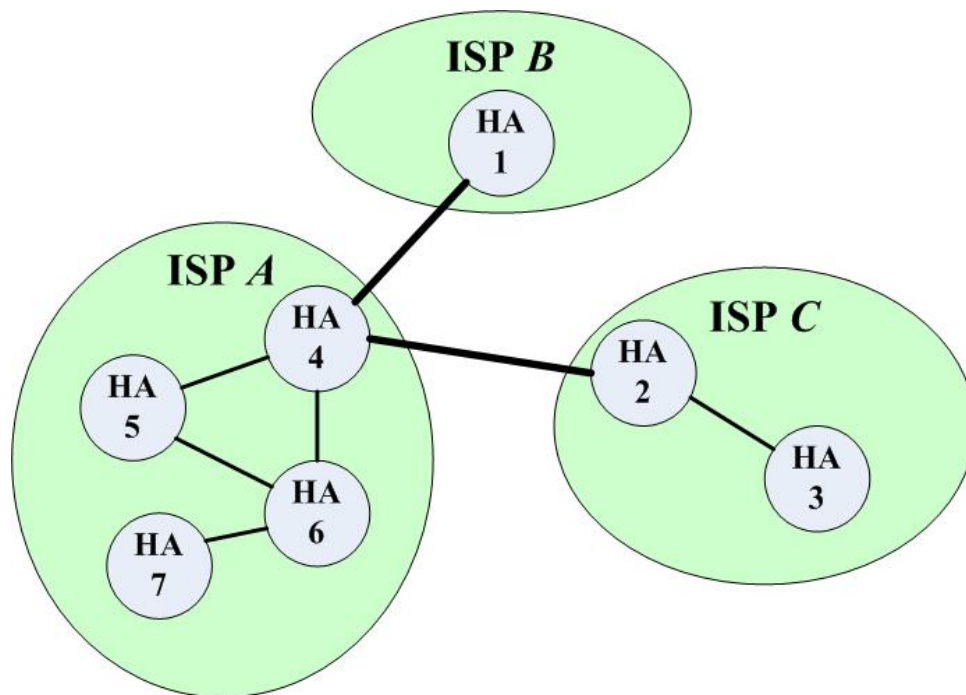
There are some assumptions in this protocol; these assumptions are described in below.

#### FHA-CA Assumption

1. There are some trust relationships between some home agents. If a home agent trusts another home agent, the trusted home agent is named Friend Home Agent (FHA).
2. A home agent may trust some local CAs.
3. Traffic between the mobile node and the home agent is secure. Integrity and Confidentiality must be fulfilled.



In the proposal protocol, the home agent has trust relationship with some friend home agents and local CAs. A local CA is one of CA parties, but not global necessarily. From the specification of MIPv6, the home agent always installs in a router. An Internet Service Provider (ISP) always has several routers, some of them with the home agent function. The trust relationship of home agents under the same ISP is easy to establish. But the trust relationship of home agents cross ISP is established by manual process. Similar concept in discussed in [19]. Figure 3-2 shows these relationships.



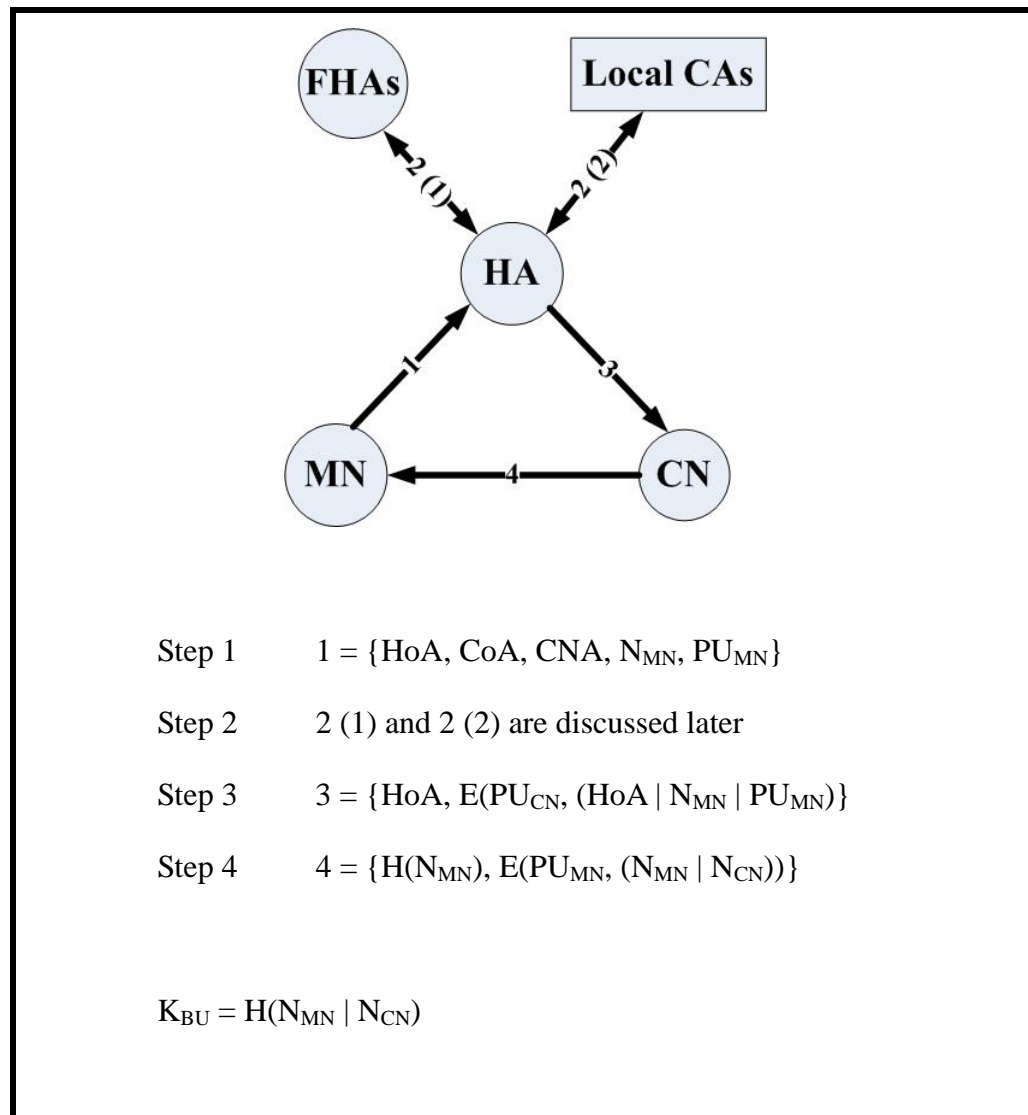
**Figure 3–1 Trust Relationship of HAs between Different ISPs**

A home agent has a FHA list recording all friend home agents. A home agent can trust other home agent through its friend home agents by checking FHA list. In figure 3-2, HA1 can trust HA3 through HA4 and HA2. If HA1 was authenticated by HA2 and HA2 was authenticated by HA1 through HA4, then HA1 adds HA2 into its FHA list. In this way the most home agents in the world

can trust each other. There is an important issue that every home agent has a responsibility for the nodes of its supporting mobility.

A home agent has a visited cache which records visited node's HoA, public key and expire time. Visited cache has a maximum size. If numbers of the entities in the cache is maximal, the home agent won't add new entity into the cache until some entity is expired and removed from the cache.

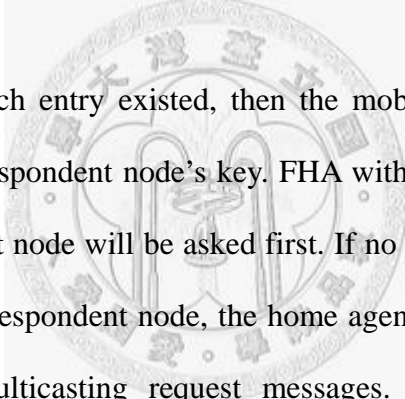
The whole process of FHA-CA protocol is shown in figure 3-1.



**Figure 3–2 Architecture of FHA-CA Protocol**

We now explain the process of the protocol. The following are the steps that lead to establishment of a shared key for binding update between the mobile node and the correspondent node.

1. The mobile node sends his home address, care-of address, address of the correspondent node, a new generated random number  $N_{MN}$ , and his public key to the correspondent node.
2. While the home agent receive step 1 message, he checks the visited cache to search the public key of the correspondent node by address of the correspondent node.



If there is no such entry existed, then the mobile node asks his FHA to retrieve the correspondent node's key. FHA with the same subnet prefix of the correspondent node will be asked first. If no FHA with the same subnet prefix of the correspondent node, the home agent will ask other FHA from FHA list by multicasting request messages. Because there are some mechanisms for HAs to certificate each other (e.g., IPSec), the exchange of the correspondent node' public key will be protected by these mechanisms. The home agent can trust the key exchange from FHA is correct.

If there is no records existed in any FHA, the home agent can ask his trusted local CA to retrieve the correspondent node's public key. The worst case is no records existed in any FHA or trusted local CA and then the home agent sent an acknowledgment message to inform the mobile node to change another protocol (e.g., the proposal FHA-CA protocol) to exchange key, or inform the mobile node that routing optimization mechanism should

not be used.

3. After retrieving the public key of the correspondent node, the home agent stores the public key in his visited cache if visited cache is still of capacity. Then, the home agent encrypts concatenation of HoA,  $N_{MN}$  and the mobile node's public key, and sends the ciphertext to the correspondent node.
4. The correspondent node decrypts the ciphertext by his private key. If HoA from the ciphertext is the same as HoA in the message, then the correspondent node can verify validity of this message. The correspondent node randomizes a number  $N_{CN}$ , hashing  $N_{MN}$ , and encrypts concatenation of  $N_{MN}$  and  $N_{CN}$  by public key of the mobile node. Afterward, the correspondent node calculates the  $K_{BU}$  and sends the message to the mobile node.

While the mobile node received the step 4 message, he checks his binding update list to conform step 1 message was sent and then hashes the  $N_{MN}$  to check the validity of the hashing code in message. If both checks are correct, then the mobile node decrypts the ciphertext to get  $N_{CN}$  and calculates the  $K_{BU}$ .

## 3.3 Non-Infrastructure-based Protocol

We propose two protocols to fix some loopholes and reduce computation caused by previous studies using CGA technology. The two protocols have the same message flow, but different content of messages.

### 3.3.1 RSADH Protocol

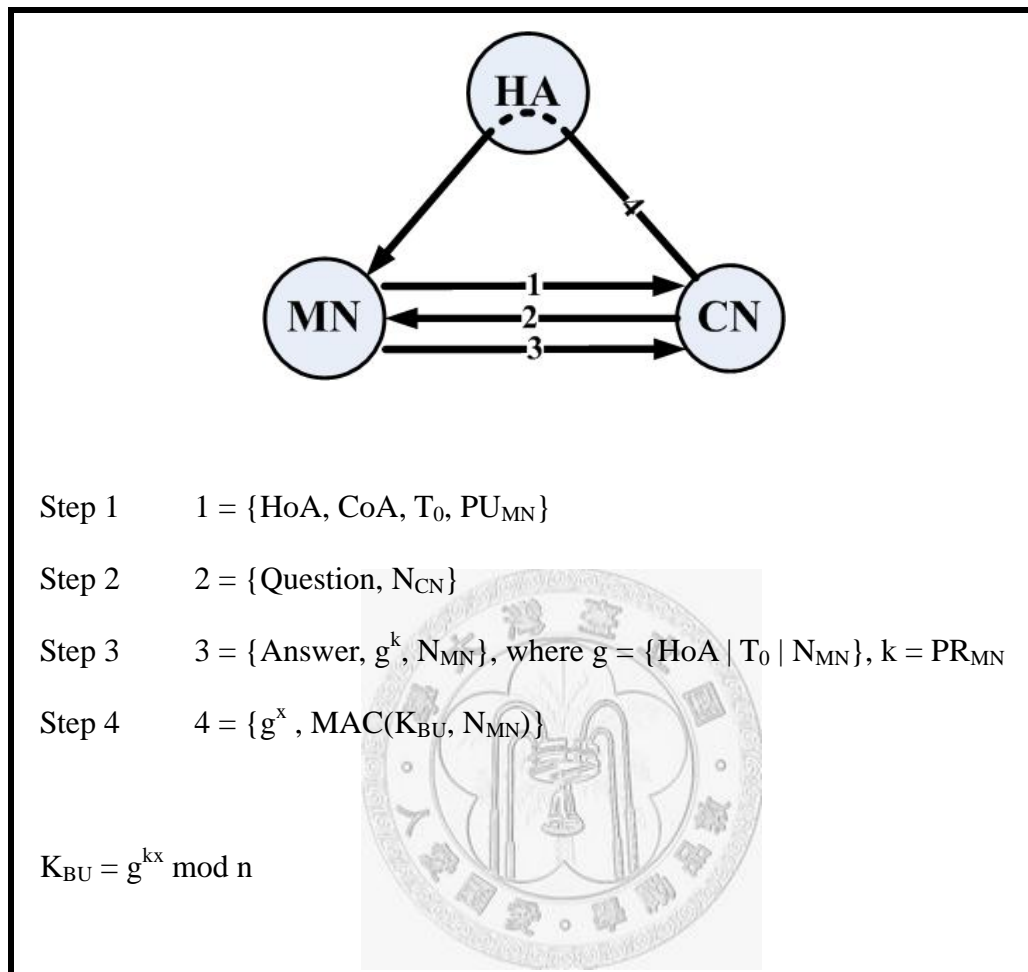
CAM-DH and SUCV use CGA and DH to exchange the key of binding update. In Chapter 2, some loopholes in these two protocols were surveyed. In RSADH protocol, these loopholes are fixed to defend threats listed in Chapter 2 except MITM treat. On the other hand, we combine DH and RSA tiredly to reduce computation in the protocol. Performance and security analysis will be discussed in Chapter 4.

#### Design Goals

1. Follow the basic assumption as shown in table 2-1.
2. Combine the RSA and DH protocols to reduce the complexity of computation.
3. Remove the loopholes in other existed protocols with CGA except the MITM attack.
4. Protocol with scalability.

## Protocol Operation

The whole process of RSADH protocol is shown in figure 3-3.



**Figure 3–3 Architecture of RSADH Protocol**

We now explain the process of the protocol. The following are the steps that lead to establishment of a shared key for binding update between the mobile node and the correspondent node.

1. The mobile node sends his home address, care-of address, timestamp, and his public key to the correspondent node.
2. While the correspondent node received the message, he checks the public key's validity. If the public key is valid, then the public key is stored.

Afterward, he generates a random number  $N_{CN}$  and a question. Question's format is discussed in Table 3-1. The correspondent node sends a question plus  $N_{CN}$  to the mobile node.

**Table 3-1 Structure of a Question and an Answer**

<p><b>Question:</b></p> <p>The question is “Is <math>n</math> a quadratic residue modulo <math>p</math>?”</p>				
<table border="1"> <tr> <td>Question Format</td> </tr> <tr> <td><math>\{n, p\}</math>, where <math>n</math> is a random number smaller than <math>p</math> but greater than 1, <math>p</math> is a prime number.</td> </tr> </table>	Question Format	$\{n, p\}$ , where $n$ is a random number smaller than $p$ but greater than 1, $p$ is a prime number.		
Question Format				
$\{n, p\}$ , where $n$ is a random number smaller than $p$ but greater than 1, $p$ is a prime number.				
<p><b>Answer:</b></p> <p>If <math>n</math> is a quadratic residue modulo <math>p</math>, then return the root of <math>n</math> –<math>r</math> –back and set a flag 1 to indicate that answer is true.</p> <p>Otherwise, return <math>p</math> back and set a flag 0 to indicate that answer is false.</p>				
<table border="1"> <tr> <td>Answer Format</td> </tr> <tr> <td><math>\{1, r\}</math>, if <math>n</math> is a quadratic residue modulo <math>p</math>, <math>r</math> is the root of <math>n</math>.</td> </tr> <tr> <td>Or</td> </tr> <tr> <td><math>\{0, p\}</math>, if <math>n</math> is a quadratic non-residue modulo <math>p</math></td> </tr> </table>	Answer Format	$\{1, r\}$ , if $n$ is a quadratic residue modulo $p$ , $r$ is the root of $n$ .	Or	$\{0, p\}$ , if $n$ is a quadratic non-residue modulo $p$
Answer Format				
$\{1, r\}$ , if $n$ is a quadratic residue modulo $p$ , $r$ is the root of $n$ .				
Or				
$\{0, p\}$ , if $n$ is a quadratic non-residue modulo $p$				

3. After mobile node received step 2 message, he checks his binding update list to make sure that he had send the request to the correspondent node. If not such record exists, discard this message. Otherwise, the mobile node tries to solve the question contained in the message and generate a new generator  $g$  as shown in table 3-2. Once  $g$  is generated, calculate a number  $N_{MN}$  by following format:  $g = \{HoA | T_0 | N_{MN}\}$ .

After above step was finished, the mobile node uses his private key to sign  $g$ .  $g$  is signed by following computation like message signed in RSA.

Sign:  $g^{PR_{MN}} \bmod n$ , where  $n$  is the parameter of RSA.

The mobile node sends a message containing the answer of the question from the correspondent node, signed  $g$  and  $N_{MN}$  to the correspondent node.

4. The correspondent node received the message from the mobile node. He checks the correctness of the answer from the mobile node. If the answer is correct, he uses stored public key to verify the validity of  $g$ .  $g$  is verify as following:  $(g^{PR_{MN}})^{PU_{MN}} \bmod n = g$  (proved by RSA).

The correspondent node knows  $HoA$ ,  $T_0$ , and  $N_{MN}$ ; he uses them to verify  $g$ . If  $g$  is correct, then he randomize a number  $x$  and use  $x$  to generate  $K_{BU}$ . After  $K_{BU}$  is generated, he calculates the MAC by  $K_{BU}$  and sends this MAC and  $g^x$  to the location of home address.

If the home address is correct, the home agent will receive the step 4 message and relays it to the mobile node. The mobile node receives the message sent from the home agent, he generate  $K_{BU}$  first and use  $K_{BU}$  to calculate the MAC. If the calculated MAC is not the same with MAC in message, the mobile node



discards the message and halts this protocol, or else the mobile node accepts the  $K_{BU}$  and uses it in the following binding update process.

**Table 3–2 Generator  $g$**

### **Process of Generator $g$ Generating**

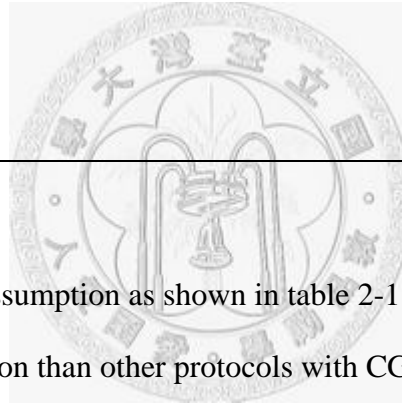
1. Let  $p', q'$  are prime numbers, such that  $p=2*p'+1$ ,  $q=2*q'+1$  and  $p, q$  are prime numbers.  $n=p*q$  ( $n$  is RSA system parameter)
2. Random pick  $a \in Z_p^*, b \in Z_q^*$
3. Set  $g = a^{(p-1)/p'} \pmod{p} = a^2 \pmod{p}$  and  $g = b^{(q-1)/q'} \pmod{q} = b^2 \pmod{q}$
4. By Chinese Remainder Theorem,  $g$  can be calculated and is unique
5. If  $g$  is generated before, then  $g$  can be generated as following :  
 $g' = g^r$ , where  $r$  is a very small integer (e.g., 2 or 3),  $r, p', q'$  are co-prime.

### **Properties of Generator $g$**

1. Let  $\langle g \rangle = \{x^2 \pmod{n} : x \in (Z_n^*)\}$ , size of  $\langle g \rangle$  ( $\#\langle g \rangle$ ) =  $(p-1)(q-1)/4 = p'q'$  [16].
2. Because  $\#\langle g \rangle = p'q'$ ,  $\langle g \rangle$  contains  $(p'-1)(q'-1)$  generators [9].

### 3.3.2 HNK Protocol

CAM-DH and SUCV use DH to protect eavesdropping attack. Based on discrete logarithm problem, even if the adversary knows the content of message, it is difficult to retrieve the  $K_{BU}$ . But the adversary breaking the discrete logarithm problem can get  $K_{BU}$ . On the other hand, these two protocols use asymmetric key algorithm to authenticate the messages. Hashing Nonce Key Protocol (HNK) try to use asymmetric key algorithm to authenticate the messages and encrypt necessary information. The difficult for breaking this protocol is the same with DH or little easier, it depends on what asymmetric key algorithm is chosen. Performance and security analysis will be discussed in Chapter 4.



#### Design Goals

1. Follow the basic assumption as shown in table 2-1.
2. Use less computation than other protocols with CGA.
3. Remove the loopholes in other existed protocols with CGA except the MITM attack.
4. Protocol with scalability.

## Protocol Operation

The whole process of HNK protocol is shown in figure 3-4.

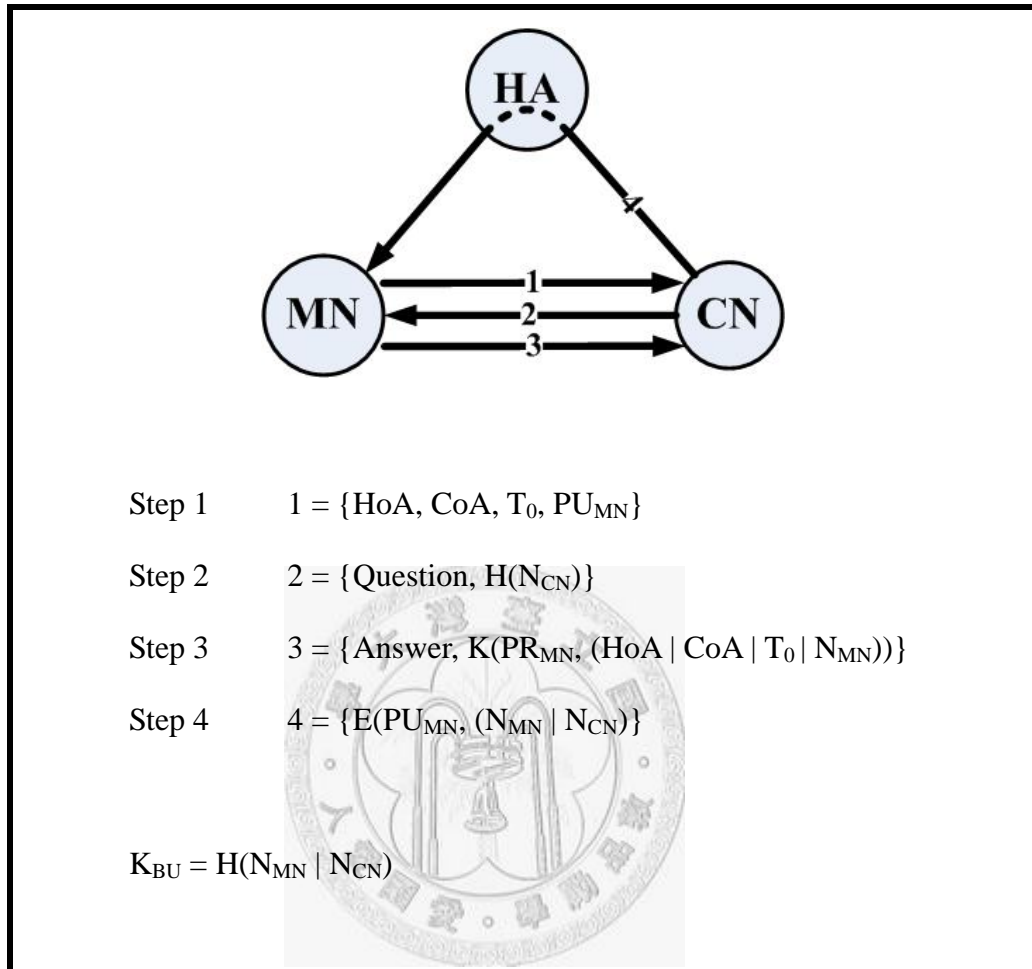


Figure 3–4 Architecture of HNK Protocol

We now explain the process of the protocol. The following are the steps that lead to establishment of a shared key for binding update between the mobile node and the correspondent node.

1. The mobile node sends his home address, care-of address, timestamp, and his public key to the correspondent node.
2. While the correspondent node received the message, he checks the public key's validity. If the public key is valid, then the public key is stored.

Afterward, he generates a random number  $N_{CN}$  and a question. Question's format is discussed in Table 3-1. The correspondent node sends a question plus hashing  $N_{CN}$  to the mobile node.

3. After mobile node received step 2 message, he checks his binding update list to make sure that he had send the request to the correspondent node. If not such record exists, discard this message. Otherwise, the mobile node tries to solve the question contained in the message, generate a random number  $N_{MN}$ , and encrypt the HoA, CoA,  $T_0$  and  $N_{MN}$  by his private key. At last, the mobile node sends a message containing answer of the question from the correspondent node and the ciphertext to the correspondent node.
4. The correspondent node received the message from the mobile node. He checks the correctness of the answer from the mobile node. If the answer is correct, he uses stored public key to decrypt the ciphertext and verify HoA, CoA, and  $T_0$ . The correspondent node uses hash function to generate  $K_{BU}$  if above check is correct. Afterward, the correspondent node uses the mobile node's public key to encrypt concatenation of  $N_{MN}$  and  $N_{CN}$  and send it to the location of home address.

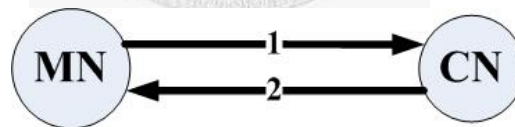
If the home address is correct, the home agent will receive the step 4 message and relays it to the mobile node. The mobile node receives the message sent from the home agent, and decrypts the ciphertext by his private key. If  $N_{MN}$  is not modified, then the mobile node trust  $N_{CN}$  and generate KBU by hashing concatenation of  $N_{MN}$  and  $N_{CN}$ .

### 3.4 Binding Update Process

In the specification of MIPv6, the lifetime of a RR authorized binding is a maximum of 420 seconds [6]. If  $K_{BU}$  has to be renegotiated every 7 minutes, it will increase a lot of load by applying protocol of intensive computation such as CAM-DH or the proposal protocols. For the reason of reducing computation and numbers of messages, we negotiate  $K_{BU}$  while the lifetime of  $K_{BU}$  is expired by sending a BU to the correspondent node.

**Forward secrecy:** The confidence that the compromise of a long-term private key does not compromise any earlier session keys.

To achieve the goal of forward secrecy, we use DH to secure binding update and key exchange with it. The whole process of binding update is shown in figure 3-5.



#### Binding Update with Key Exchange

Step 1  $1 = \{BU, g^x, E(K_{BU}, (H(BU | g^x) | N_{MN} | T))\}$ , T is expire time

Step 2  $2 = \{BA, g^y, E(K_{NBU}, (H(BA | g^y) | N_{MN} | T))\}$ , T is expire time

$K_{NBU} = g^{xy} \text{ mod } p$ , where p is a prime number. g and p are system parameters .

### Binding Update when Binding Update Key is not expired yet

Step 1  $1 = \{BU, MAC(K_{BU}, BU)\}$

Step 2  $2 = \{BA, MAC(K_{BU}, BA)\}$

**Figure 3–5 Process of Binding Update**

We now explain the process of binding update with key exchange.

1. The mobile node generates a binding update message (BU) defined in [6].  
If  $K_{BU}$  is expired, the mobile node generates a  $g^x$  where  $x$  is a random number smaller than  $p$  and then hash the concatenation of BU and  $g^x$ . The whole new binding update message will contain BU,  $g^x$ , and a ciphertext. The ciphertext is encrypted from hashing the concatenation of BU and  $g^x$ , a random number  $N_{MN}$  and expire time  $T$  by  $K_{BU}$ .
2. While the correspondent node receives the binding update, he uses  $K_{BU}$  to decrypt the ciphertext and retrieves hashing code and  $N_{MN}$ . The correspondent node verifies the validity of the binding update by hashing the concatenation of BU and  $g^x$ . If the binding update is valid, the correspondent node randomizes a new number  $y$  and uses it to calculate new binding update key  $K_{NBU}$ .

After  $K_{NBU}$  is calculated, the correspondent node generate a binding acknowledgement (BA) defined in [6]. He uses similar way in step 1, but substitutes BU for BA, substitutes  $K_{NBU}$  for  $K_{BU}$  and substitutes  $g^y$  for  $g^x$ . This binding acknowledgement message is sent to the mobile node.

The mobile node checks his binding update list first after he received the binding acknowledgement. Next step, he calculate new binding update key  $K_{NBU}$  to decrypt the ciphertext in the binding acknowledgement. If  $N_{MN}$  and hashing code is correct, the mobile node accepts  $K_{NBU}$  and stores  $K_{NBU}$  and  $T$ .



## Chapter 4 Evaluation

In this chapter, we will show the performance improvement in the non-infrastructure-based protocol. The computation complexity in asymmetric key algorithm and size of packets for the RSADH protocol are less than the CAM-DH and SUCV protocol. The HNK protocol is superior in performance than other CGA protocols.

Threats in binding update of the proposal protocols will be assessed. The FHA-CA protocol is immune to these threats. Besides, some loopholes in other CGA or lightweight protocols are not existed in the proposal non-infrastructure-based protocols except the unsolved MITM attacks.

### 4.1 Assessment of Computational Intensity

#### 4.1.1 Performance of RSADH Protocol

The RSADH protocol provides better performance in computation than the CAM-DH and SUCV protocol. The DH protocol or asymmetric key algorithm cost much more time than normal computation or symmetric key algorithm.



Hence, reduce the numbers of computation in DH or asymmetric key algorithm will improve the performance well.

In the CAM-DH and SUCV protocols, there are six intensive computations as shown in table 4-1. But in the RSADH, we combine the DH and the RSA tiredly to reduce one intensive computation. For the mobile node, step 1 and step 2 in table 4-1 are combined into one step, and time of step 1 is reduced. The intensive Computation of RSADH is shown in Table 4-2. On the other hand, the  $g$  generating is low computation because the power of  $g$  is small. Hence, time for  $g$  generating can't compare to DH value generating.

**Table 4-1 Six Intensive Computations in CAM-DH and SUCV**

<p>For the MN</p> <ol style="list-style-type: none"><li>1. Generate DH value (<math>g^x \text{ mod } p</math>)</li><li>2. Sign the message by private key</li><li>3. Calculate the binding update key (<math>g^{xy} \text{ mod } p</math>)</li></ol> <p>For the CN</p> <ol style="list-style-type: none"><li>1. Generate DH value (<math>g^y \text{ mod } p</math>)</li><li>2. Verify the message by public key</li><li>3. Calculate the binding update key (<math>g^{xy} \text{ mod } p</math>)</li></ol>
--

**Table 4–2 Five Intensive Computations in RSADH Protocol**

<p>For the MN</p> <ol style="list-style-type: none"><li>1. Sign the <math>g</math> by private key, <math>g</math> can be view as a plaintext in RSA.</li><li>2. Calculate the binding update key (<math>g^{kx} \bmod p</math>)</li></ol> <p>For the CN</p> <ol style="list-style-type: none"><li>1. Generate DH value (<math>g^x \bmod p</math>)</li><li>2. Verify the <math>g</math> by public key</li><li>3. Calculate the binding update key (<math>g^{kx} \bmod p</math>)</li></ol>
---

In the SUCV protocol, RSA algorithm is used, and the DH value is at least 1536 bits equal to the length of RSA signature. Therefore, the size of  $g$  in the DH is equal to  $n$  in the RSA. In the RSADH protocol, the size of  $g$  is also equal to  $n$  in the RSA as shown in table 3-2.

The computations of the DH protocol of the SUCV are  $O(m^x)$  where  $m$  is  $2^{1536}$  and  $x$  is not specified in SUCV but smaller than  $m$ . The computations of the RSA protocol of the SUCV are  $O(m^m)$  where  $m$  is  $2^{1536}$ . Thus, the total computations of the SUCV are  $4*O(m^x) + 2*O(m^m)$ . On the other hand, the computations of the DH protocol of the RSADH are  $O(mx)$  where  $m$  is  $2^{1536}$  and we let  $x$  is equal to the size of the SUCV. The computations of the RSA protocol of the RSADH are  $O(m^m)$  where  $m$  is  $2^{1536}$ . Thus, the total computations of the RSADH are  $3*O(m^x) + 2*O(m^m)$  smaller than SUCV.

The combination of DH and RSA tiredly also reduces the size of packet. In

table 4-1, step 1 and step 2 produce two big values in case of brute force attacks. However, the value of step 1 is removed from the RDADH protocol, and it reduces the total size of packets.

### 4.1.2 Performance of HNK Protocol

We remove DH in the HNK protocol. The numbers of intensive computations are only two and size of packets can also be reduced. It provides the better performance than other CGA protocols.

## 4.2 Assessment of Security

We summarize the assessment of security in table 4-3 were detailed analysis will be discussed in Sections 4.1 and 4.2.

**Table 4–3 Security Analysis of the Proposal Protocols**

<b>Security Analysis</b>					
<i>Protocol</i>	<i>Passive Attack</i>	<i>Active Attacks</i>			
	Eavesdrop Attack	Stealing Traffic Attack	Reflection Attack	MITM Attack	DoS Attack
FHA-CA	X	X	X	X	√
RSADH	X	X	X	√	X
HNK	X	X	X	√	X
<p>√: It is not hard for the adversary to use this attack on the protocol</p> <p>X: The protocol can defend against this threat well</p>					

## 4.2.1 Security of the FHA-CA Protocol

The FHA-CA protocol provides a good security level by authentication between the mobile node and the correspondent node. But it is difficult to defend against DoS attacks because the mobile node requests the correspondent node's public key every initial of process. We discuss the issue for FHA-CA protocol to defend against the threats.

The nonce of the mobile node and the correspondent node is protected by their public key, thus it makes eavesdrop attack difficult.

### 1. Stealing Traffic Attack

Because the binding update key is protected by public keys of the mobile node and the correspondent node, it is difficult for adversary to retrieve the binding update key. If the adversary didn't possess the binding update key, he can't send a forged binding update to steal the traffic.

### 2. Reflection Attack

While the correspondent node received the message from the home agent, he decrypts the ciphertext in the message by the public key of the mobile node and encrypts a message, then sent it to the mobile node. After the correspondent node receives a message, he sends a message out. It won't cause the binding update storm. Thus, the binding update can't be used to flooding victims.

### 3. MITM Attack

Under the support of CA, the MITM attack can be defended in this protocol. Every network node has to identify himself to run the protocol. If some node can't identify himself, the route optimization mechanism will be stopped and the binding update process will be only exchange between the home agent and the correspondent node.

### 4. DoS Attack

While the correspondent node received the message from the home agent, he will decrypt the ciphertext and encrypt another message. The adversary can use a lot of forged binding update to run out of the resource of the correspondent node.

## 4.2.2 Security of the RSADH and HNK Protocol

As shown in table 4-3, a summary of security of proposal protocols lists the possible attacks on the protocols. In this section, more details will be discussed by different protocols in non-infrastructure-based protocol.

### 1. Stealing Traffic Attack

#### **RSADH protocol:**

If the adversary can't change initial round of protocol to hide in the middle of the mobile node and the correspondent node, it is difficult to retrieve the binding update key for adversary. Because the DH and RSA are hard problem, the adversary can't get the binding update key easily. Without this binding update key, the adversary can't authenticate his forged binding

update and use it to steal traffic.

### **HNK protocol:**

This protocol uses asymmetric key algorithm to protect the binding update key. If the length of the public/private key is long enough, then the binding update key is difficult to be retrieved. Without this binding update key, the adversary can't authenticate his forged binding update and use it to steal traffic.

## 2. Reflection Attack

### **RSADH protocol and HNK protocol:**

In the both protocols, while the correspondent node received the message from the mobile node, he sent a message back to the original mobile node. If some adversaries use forged binding update to attack the correspondent node, the adversaries consume the same resource. Several protocols can be used to flooding attack by amplifying the binding update, but not exist in the RSADH or HNK protocol.

## 3. MITM Attack

### **RSADH protocol and HNK protocol:**

As describe in the Section 2.3, MITM attack only can be defend in pre-shared secret protocol such as PKI-based protocol. In this two protocols, MITM attack is still unavoidable, but the adversary must modify all messages in the protocol than three messages in other CGA protocol (e.g., CAM-DH, SUCV, CGA). Although it provides slightly improvement on defending MITM attack but is still vulnerable.

#### 4. DoS Attack

##### **RSADH protocol and HNK protocol:**

In both protocols, we adapt a question to defend the DoS attack. Similar concept is discussed in [10]. In the step 2 of the both protocol, a question is sent to the mobile node to consume some resource of the mobile node. If the mobile nodes send a lot of messages to exhaust the correspondent node, the victim do nothing but generate two random numbers unless the mobile solved the problem. It is difficult to solve the question but is easy to check the answer. Before running out the resource of the correspondent node, the mobile node will exhaust his resource. However, the both protocols are still vulnerable in DDoS (Distributed Denial of Service).

### **4.3 Discussion**

There are two dimensions to be considered: Security and Performance. In the FHA-CA protocol, the security issue is the first consideration. Although the performance is sacrificed, the protocol provides the best security level. In practice, such infrastructure may be hard to establish. Hence, we need other alternative protocols to support security when such infrastructure is not existed. RSADH and HNK protocols are proposed under this situation. In the analysis of performance and security, HNK protocol has the same security level with RSADH but better performance. If no pre-established infrastructure is existed, HNK protocol is a good alternative solution to secure the binding update in MIPv6.

# Chapter 5 Conclusion

## 5.1 Summary

While mobile devices are used more and more, the MIP protocol needs to be developed quickly. The security issue is an important key for the development of MIP. Removing possible threats in MIP becomes the first issue for MIP designers.

To address the security issue, we discuss possible threats in binding update mechanism. These threats can be used to attack the victims or steal traffic from victims. The designers should consider these threats before they develop a protocol to securing the binding update.

Several protocols are discussed to expose the insufficiency of security in binding update. Related security issues are analyzed to show requirement of a new protocol to securing the binding update mechanism.

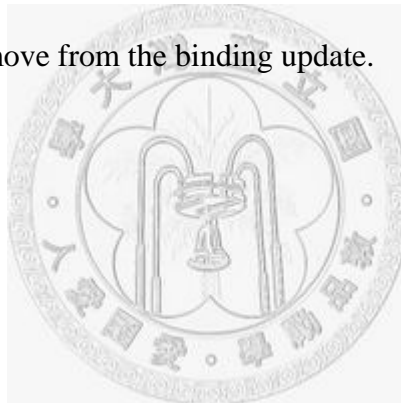
Three protocols are proposed to fix the security problem from other's protocols. These protocols are capacity of better security level and performance. Related



security issues are discussed carefully to prove the robustness of security. At the same security level, the proposal protocols with better performance are good solutions for the binding update mechanism in MIP.

## 5.2 Future Work

Year after year, better technologies were proposed to solve the security problem. There are no efficient solutions to solve MITM Problem. If CA is introduced, the computation complexity increases quickly. Thus, a good identity authentication protocol is urgent for several applications. If such a protocol is proposed, then it will solve the MITM attack in RSADH and HNK protocols and make threats remove from the binding update.



# References

- [1] J. Arkko and P. Nikander. “Weak Authentication: How to Authenticate Unknown Trusted Parties”, Proceedings of Security Protocols Workshop 2002, April, 2002
- [2] S Deering, R. Hinden, “Internet Protocol Version 6 (IPv6) Specification,” IETF RFC 2460, December 1998
- [3] R. H. Deng, J. Zhou, F. Bao, “Defending Against Redirect Attacks in Mobile IP”, CCS’02, November 2002.
- [4] R. Droms, J. Bound, B. Volz, T. Lemon, C. Perkins, M. Carney, “Dynamic Host Configuration Protocol for IPv6 (DHCPv6)”, IETF RFC 3315, July 2003
- [5] S. Jacobs, S. Belgard, “Mobile IP Public Key Based Authentication”, IETF Internet draft <draft-jacobs-mobileip-pki-auth-03.txt>, July 2001.
- [6] D. Johnson, C. Perkins, J. Arkko, “Mobility Support in IPv6”, IETF Internet draft <draft-ietf-mobileip-ipv6-21.txt>, August 2003
- [7] J. Kempf, P. C. Hwang, S. Okazaki, “Cert-BU: Certificate-based Techniques for Securing Mobile IPv6 Binding Updates”, internetworking 2003, June 2003.
- [8] A. Mankin, et al., “Threat Models Introduced by Mobile Ipv6 and Requirements for Security in Mobile Ipv6”, IETF Internet draft <draft-ietf-mipv6-scrty-reqts-02.txt>, May 2001.
- [9] W. Mao, “Modern Cryptography Theory and Practice”, Prentice Hall PTR, 2004
- [10] G. Montenegro, C. Castelluccia, “SUCV Identifiers and Addresses”, IETF Internet draft <draft-montenegro-sucv-03.txt>, July 2002.

- [11] P. Nikander, C. Perkins, “Binding Authentication Key Establishment Protocol for Mobile IPv6”, IETF Internet draft <draft-perkins-bake-01.txt>, July 2001.
- [12] G. O’Shea, M. Roe, “Child-proof Authentication for MIPv6 (CAM)”, Computer Communication Review, April 2001
- [13] S. Okazaki, A. Desai, C. Gentry, J. Kempt, Alice, Silverberg, Y. L. Yin, “Securing MIPv6 Binding Updates Using Address Based Keys (ABKs)”, IETF Internet draft <draft-okazaki-mobileip-abk-01.txt>, October 2002.
- [14] M. Roe, T. Aura, G. O’Shea, J. Arkko, “Authentication of Mobile IPv6 Binding Updates and Acknowledgments”, IETF Internet draft <draft-roe-mobileip-updateauth-02.txt>, February 2002.
- [15] P. Shrivastava, A. Murty, A. Gurtu, D. Hemwani, ”Securing Mobile IPv6 Binding Updates”, USC, September 2002.
- [16] N. Smart, “Cryptograhpy: An Introduction”, McGraw-Hill, 2003
- [17] H. Soliman, “Mobile IPv6: Mobility in a Wireless Internet”, Addison-Wesley, April 2004.
- [18] S. Thomson, T. Narten, “IPv6 Stateless Address Autoconfiguration”, IETF RFC 2462, December 1998
- [19] J. Zao, et al., ”A Public-Key Based Secure Mobile IP”, Wireless Networks, October 1999.



# 簡歷

姓 名：余俊達

出生地：台灣省台南市

出生日：中華民國六十六年十月十五日

學 歷：八十七年九月至九十一年六月

國立台灣大學資訊管理學系

九十年九月至九十二年七月

國立台灣大學資訊管理研究所