ARTICLE IN PRESS

Available online at www.sciencedirect.com
SCIENCE DIRECT®

Computer Networks xxx (2005) xxx–xxx

Computer Networks

www.elsevier.com/locate/comnet

Routing optimization security in mobile IPv6

Kui Ren [a,*], Wenjing Lou [a], Kai Zeng [a], Feng Bao [b], Jianying Zhou [b], Robert H. Deng [c]

[a] *Department of Electrical and Computer Engineering, Worcester Polytechnic Institute, 100 Institute Road, Worcester, MA 01609, United States*
[b] *Institute for Infocomm Research, 21 Heng Mui Keng Terrace, Singapore 119613, Singapore*
[c] *School of Information Systems, Singapore Management University, Singapore 259756, Singapore*

Received 30 March 2004; received in revised form 1 August 2005; accepted 17 September 2005

Responsible Editor: C. Blondia

Abstract

Route Optimization (RO) in Mobile IPv6 (MIPv6) provides a mobile node (MN) the opportunity to eliminate the inefficient triangle routing with its corresponding node (CN) and therefore, greatly improves the network performance. However, in doing so, MIPv6 introduces several security vulnerabilities, and among them a major concern is the authentication and authorization of Binding Updates (BUs) during the RO process. Unauthenticated or malicious BUs open the door for many types of attacks. As every IPv6 node is expected to support MIPv6, mechanisms to secure BU will have a significant impact on the next generation Internet. In this paper, based on an in-depth analysis of the security weaknesses existing in previously proposed protocols, a light-weight BU protocol with high security strength is proposed, which makes use of public key certificate-based strong authentication technique. Another important contribution of the paper is the introduction of a novel and scalable 3-layer trust management framework, which takes advantage of IPv6 address format and home link's jurisdiction over the addresses it assigns, and thereby solves the difficult certificate issuing and management problem presented in the previous public key certificate-based solutions via trust delegation. The proposed protocol is highly efficient in term of both computation and communication costs on both MN and CN sides. An extended protocol is also proposed to explicitly support Hierarchical MIPv6 (HMIPv6).

Keywords: MIPv6; Routing optimization; Security; Binding Update protocol

1. Introduction

Mobile IP version 6 (MIPv6) is an IP-layer mobility protocol for the IPv6 Internet [11,12,23], which is designed based on the idea of providing mobility support on top of the existing IP infrastructure, without requiring any modifications to routers, applications or stationary hosts. The MIP allows transport layer sessions (TCP connections or UDP-based transactions) to continue even if the underlying host(s) move and change their IP addresses; it also allows a node to be reached

* Corresponding author. Tel.: +1 508 831 5783.
E-mail address: kren@wpi.edu (K. Ren).

1389-1286/$ - see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.comnet.2005.09.019

through a static IP address. MIPv6 takes the view that IP addresses can be used as natural identifiers of nodes, as they have been used since the beginning of the Internet.

In MIP, a mobile node (MN) is addressed by two IP addresses, a home address (HoA) and a care-of address (CoA). A MN has its static HoA at its home subnet. When moving to a new subnet, the MN will discover the default router, perform (stateful or stateless [9,32]) address auto-configuration, and use its new address as CoA. The former is an IP address assigned to MN within its subnet prefix on its home link and the latter is a temporary address acquired by MN while visiting a foreign link. This dual address mechanism realizes the design goal of MIP. Mobile IP version 4 (MIPv4) was specified in [23] and MIPv6 was specified in [12]. Mobility support in IPv6 is considered particularly important, since mobile devices are predicted to account for a significant fraction of the population of the Internet during the lifetime of IPv6.

In MIPv4, when a MN changes location, it obtains a CoA and informs its Home Agent (HA) its new CoA and the HA encapsulates and tunnels any packets it receives for MN on its home network to this CoA. Therefore, every time a correspondent node (CN) sends a packet to MN, while MN is away from home link, the packet first travels to the home network before reaching the MN. This inefficient routing is called triangle routing. In MIPv6, there are two possible modes for communications between the MN and a CN. One mode is called bidirectional tunnelling. In this mode, packets from the CN are routed to the home agent and then tunnelled to the MN. Packets to the CN are tunnelled from the MN to the home agent (''reverse tunnelled'') and then routed normally from the home network to the CN. The other mode, ''Route Optimization''(RO), requires the mobile node to register its current binding at the correspondent node. RO provides MN the opportunity to eliminate the inefficient triangle routing and bidirectional MN–HA tunnelling as shown in Fig. 1. On receiving a tunnelled packet from its HA, MN knows that CN that sent the packet is unaware of its current CoA. MN may choose to inform CN its new CoA using a Binding Update (BU) message, thereby allow CN to send subsequent packets directly to MN. Unfortunately, unauthenticated or malicious BU messages provide intruders an easy means to launch various types of attacks [7,8,16,19,20]. Therefore, RO security is of paramount importance for MIPv6 to meet its basic security requirements.

In this paper, based on an in-depth analysis of the security weaknesses existing in previously proposed protocols, a light-weight BU protocol with high security strength is proposed. Our protocol makes use of public key certificate-based strong authentication technique. Another important contribution of the paper is the introduction of a novel and scalable 3-layer trust management framework, which takes advantage of IPv6 address format and home link's jurisdiction over the addresses it assigns and therefore, solves the difficult certificate issuing and management problem presented in the previous public key certificate-based solutions via trust delegation. Also by taking advantage of early binding update technique, the proposed protocol is highly efficient in term of both computation and communication costs with respect to both MN and CN sides. An extended protocol is also proposed to explicitly support Hierarchical MIPv6 (HMIPv6).

The rest of the paper is organized as follows: In Section 2, we give the background of security design in MIPv6, as well as the security threats. In Section 3, we review three existing state-of-the-art RO protocols presented in the IETF's Internet drafts and [10,24]. We focus on the security analysis

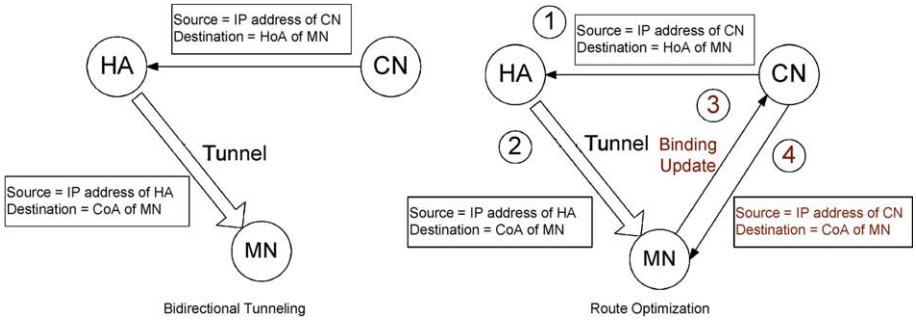


Fig. 1. Basic operations in MIPv6.

of these protocols and point out their security limitations. Section 4 is devoted to our new protocol and its analysis. Finally, Section 5 is our concluding remarks.

2. Background in MIPv6 security

2.1. Basic assumptions for MIPv6 and its security

The main assumptions [7,8,10,16,19,20] which drive the MIPv6 design and specifications are : (1) The routing prefixes available to a node are determined by its current location, and therefore the node must change its IP address as it moves; (2) The routing infrastructure is assumed to generally deliver packets to their intended destinations as identified by the destination address. The design of MIPv6 is to follow the end-to-end principle, to duly notice the differences in trust relationships between the nodes, and not to make the security any worse than IPv4 is today. The end-to-end principle is applied by restricting mobility related state primarily to the home agent. Additionally, the correspondent nodes also maintain a soft state [20]. The security assumptions made on MIPv6 are as follows:

1. Pre-established security association between a mobile node and its home agent: Mobile nodes and home agents know each other, and can thus have a pre-established strong security association to reliably authenticate exchanged messages between them. IPsec Encapsulating Security Payload (ESP) [3] is used to set up a secure tunnel between a mobile node and its home agent.
2. No pre-established security association between a mobile node and a random correspondent node: It is expected that MIPv6 will be used on a global basis between nodes belonging to different administrative domains, hence building a global authentication infrastructure to authenticate mobile nodes and random correspondent nodes would be a very demanding task, at least in the near to medium terms. Furthermore, making a traditional authentication infrastructure to keep track of correct IP addresses for all hosts is either impossible or at least very hard due to the dynamic association between IP addresses and hosts [10].

The current MIPv6 RFC reacts to the second assumption with a so-called infrastructureless authentication mechanism (i.e., Return Routability (RR) [7,8,20] or Cryptographically Generated Addresses (CGA) [5,6,29]) and thus sacrifices security strength and is exposed to certain attacks. The solution reflects only a straightforward response to the second assumption. While it is true that neither the existence nor the deployment of global public key infrastructure (PKI) (or other global security infrastructure) for authentication on MIPv6 nodes without pre-relationship is possible in the near future, this does not rule out the possibility of having fragmented authentication infrastructures within individual administration domains or even cross different domains. For example, secure BGP (*S-BGP*) relies on a PKI where announced prefixes exchanged among ASes are certified [34,35]. We also notice that IPv6 addresses are assigned in a hierarchical manner [17]. Very few blocks of addresses will be directly assigned by Internet addressing authorities. In general, IP addresses are required to obtain from its upstream Internet Service Provider (ISP), which in turn gets address from the next level up. Only the Top Level Aggregators (TLA) get addresses assigned by an addressing authority, and usually belong to global or "Tier-1" ISPs (e.g., UUNET, Sprint, AOL, AT&T, NTT, etc.) [17]. These Tier-1 ISPs cooperate extensively with each other and have well-established long term trust relationships, and hence it is easy to prove each other on the ownership of their corresponding subnet prefixes through mutually trusted certificates or other means. This observation leads to a reasonable reduction that reduces the difficult authentication problem of MIPv6 nodes to a much easier one. The problem of authentication of a MIPv6 node can firstly be delegated to the home agent it belongs to based on the first assumption. The problem then can be further delegated to the upper level ISPs the home agent belongs to. Finally, a Tier-1 ISP is supposed to authenticate itself on behalf of the mobile node. Obviously, the above reduction leads to a much simpler and more practical solution for authentication of MIPv6 nodes and therefore, allows strong security designs in MIPv6.

On the other hand, portable devices with constraint computational ability and battery life, such as PDAs and cellular phones, are predicted to account for a majority or at least a substantial fraction of the population of mobile devices [10]. Hence, computational cost at mobile nodes side should be light-weight, and public key cryptosystem operations should be avoided whenever possible in the security design. Usually, there are two ways to

reduce the computation cost: (1) Reduce the number of computationally expensive operations. (2) Delegate the computationally expensive operations to more powerful nodes (e.g., home agents in our case). The number of communication rounds is another consideration in the light-weight security design. Due to the dynamic and error-prone nature in wireless environments, less communication rounds are preferred to ensure the success of the protocol besides the consideration of saving power.

2.2. Security threats in MIPv6

The goal in designing MIPv6 is simply to make IPv6 mobile and at least as secure as the static IPv6. However, MIPv6 does introduce several additional security vulnerabilities into IPv6 and among them the biggest concern is the weak authentication and authorization of BUs. As discussed before, RO in MIPv6 greatly improves the efficiency of routing by allowing the shortest communications path to be used and eliminating congestion at the mobile node's home agent and home link; however, it also greatly increases the number of BUs sent by a MN to its CNs, and in doing so, greatly increases the security risk of MIPv6. As every IPv6 node is expected to be deployed as a MIPv6 node as well, and every MIPv6 node is to be a CN, BU security threats can been seen as applicable to the whole Internet.

Firstly, unauthenticated or malicious BUs opens the door for many types of attacks as discussed below [4,7,8,10,16,19,20,31].

False Binding Update Attack: Spoofed BUs may be sent to HAs and CNs. By spoofing BUs, an attacker can redirect traffic to itself or another node and prevent the original node from receiving traffic destined to it. For example, let us say nodes A and B have been communicating with each other, then, an attacker, node C, sends a spoofed BU packet to node B, claiming to be node A with a CoA of node C. This would cause node B to create a binding for node A's CoA and subsequent further traffic to node C, believing it to be node A's new CoA. Node A would not receive the data it was intended to receive, and, if the data in the packets is not cryptographically protected, node C will be able to see all of node A's sensitive information.

Man-in-the-Middle Attack: An attacker may also spoof BUs to two CNs in order to set itself as a Man-in-the-Middle between a MN and a CN. For example, if node A and node B are communicating, the attacker could send both nodes a spoofed BU with the CoA set to its own address. This would cause both nodes A and B to send all packets to node C rather than to each other, and therefore, results in attacks against secrecy and integrity. (Note that without RO, an attacker would have to be in the path between the nodes in order to capture and read packets.)

Denial-of-Service Attack: By sending spoofed BUs, an attacker could also send large amounts of unwanted traffic to overwhelm the resources of a single node or those of a network. The attacker could first find a site with streaming video or another heavy data stream and establish a connection with it. Then it could send a BU to the corresponding node, saying to redirect subsequent data traffic to the attacker's new location, that of an arbitrary node. This arbitrary node would be then bombed with a large amount of unnecessary traffic. Similarly, the attacker could also use spoofed BUs to redirect several streams of data to random addresses with the network prefix of a particular target network, thereby congesting an entire network with unwanted data.

Secondly, the adversary may also try to attack the BU protocol itself, and thus prevent the protocol participants from correctly completing the protocol [4,10]. Basically, this type of attacks can be identified as DoS attacks. The stateful protocols are known to expose the protocol participants to DoS attack. "In particular, if a host stores a state in a protocol run that someone else has initiated and before authenticating the initiator, an attacker can initiate the protocol many times and cause the host to store a large number of unnecessary protocol states" [7]. Other attacks of this type include reflection and amplification attack and replay attack [7,36]. In reflection and amplification attack, packets are sent into a looping path to the target (amplification); the attacker can also hide the source of a packets by reflecting the traffic from other node (reflection), and therefore, the protocol participant nodes can be tricked into sending many more packets than they receive from the attacker. In replay attack, the attacker captures the BUs of MN, and tries to replay back after MN moved away whenever possible.

Note that the different attacks assume different conditions and requirements of the attackers and therefore, the security threats vary largely. We will discuss them along with the analysis of the existing protocols.

3. Security analysis of the state-of-the-art protocols

In this section we first discuss the latest IETF Mobile IPv6 draft, i.e., Mobility Support in IPv6 [12] and its security flaws. Our certificate-based Binding Update (CBU) protocol, which was proposed among the first to improve the security strength of the current Internet draft [10], is then discussed. Comments on security limitations of the CBU protocol conclude this section. We list below the cryptographic notation used throughout the paper for easy reference:

$h(\,)$	a cryptographic secure one-way hash function, or one-way hash function in short, such as *MD5* [26] and *SHA* [22].
$prf(k, m)$	a keyed one-way pseudo-random function—often a keyed hash function [15]. It accepts a secret key k and a message m, and generates a pseudo-random output. This function is used for both generation of message authentication codes and derivation of cryptographic keys.
P_X/S_X	public/private key pair of node X in a digital signature scheme such as *RSA* [27] or *DSS* [21].
$S_X(m)$	node X's digital signature on a message m using the private key S_X.
$m\|n$	concatenation of two messages m and n.
K_{CN}	a secret key for a correspondent node—it stays the same between protocol runs, but can change periodically.

3.1. IETF related secure binding update protocols

In this subsection we describe and analyze two kinds of protocols for authenticating Binding Update messages, the Return Routability (RR) protocol appeared in [12,19] and the CGA-based protocols proposed in [14,28].

3.1.1. The RR protocol and its analysis

The RR protocol consists of two checks, a home address check and a care-of address check. Basically, it means that a node verifies that there is a node that can respond to packets sent to a given address. It is assumed that a successful reply indicates that there is indeed a node at the given address, and that the node is willing to reply to the probes sent to it. The packet flow is depicted in Fig. 2. The real return routability checks are the message pairs (Home Test, BU) and (Care-of Test, BU). The Home Test Init (HoTI) and Care-of Test Init (CoTI) packets are only needed to trigger the test packets, and the BU message acts as a combined routability response to both of the tests.

1. *Home Address check*: The Home Address check consists of a Home Test (HoT) packet and a subsequent BU. The HoT is assumed to be tunnelled by the HA to the MN. The HoT contains a cryptographically generated token, $home\ token = h(K_{cn}|HoA|n_i|0)$, which is formed by calculating a hash function over the concatenation of a secret key K_{cn} known only by the CN, the source address of the HoTI packet, and a nonce n_i. The index i is also included in the HoT packet, allowing the correspondent node to easier find the appropriate nonce.
2. *Care-of Address check*: From the CN's point of view, the care-of address check is very similar to the Home address check, but the packet is sent directly to MN's CoA. Furthermore, the token is created in a slightly different manner in order to make it impossible to use home tokens for care-of tokens or vice versa ($care\text{-}of\ token = h(K_{cn}|CoA|n_j|1)$).
3. *Forming the first Binding Update*: When the mobile node has received both the HoT and CoT messages, it creates a binding key K_{bm}: $K_{bm} = h(care\text{-}of\ token|home\ token)$ by taking a hash function over the concatenation of the tokens received. This key is used to protect the first and the subsequent binding updates, as long as the key remains valid. Note that K_{bm} is available to anyone that can obtain both CoT and HoT messages.

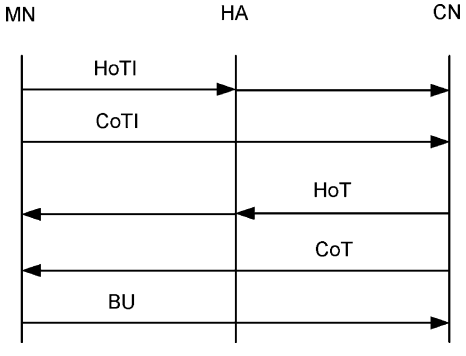


Fig. 2. The Return Routability Protocol.

In the RR protocol, the two token exchanges verify that MN is alive at its corresponding HoA and CoA, respectively. The liveness test is used to substitute authentication of nodes as the response to the infrastructureless assumption. While it is very useful to make sure the liveness of MN on both HoA and CoA, the security of RR protocol obviously hinges on the secret sharing of K_{bm} between MN and CN, which in turn hinges on the secrecy of one of the two tokens. As pointed out above, the two tokens are available to anyone that can obtain both CoT and HoT message. Hence, the security of RR protocol is considerably weak. Although the authors argued that the motivation for designing the RR protocol was to have sufficient support for MIP, without creating major new security problems. It was not the goal to protect against attacks that were already possible before the introduction of IP mobility [7,12,20]. The protocol does not defend against an intruder who can monitor the CN–HA path. Such intruders would in any case be able to mount an active attack against MN when it is at its home location. However, the design principle of the RR protocol, i.e., defending against intruder who can monitor the CN–MN path but not the CN–HA path, is fundamentally flawed since it violates the well known "weakest link" principle in security as pointed out in [10]. After all, one has no reason to assume that an intruder will monitor one link and not the other, especially when the intruder knows that monitoring a given link is particularly effective to expedite its attack. While it is true that intruders are able to mount active attacks when a node is at home in the base IPv6, it is much easier to launch such attacks in MIPv6 than in the base IPv6.

First, let us consider the false BU attack. In the case of the base IPv6 without mobility (which is equivalent to the mobile node MN at its home link in MIPv6), to succeed in the attack, the intruder must be constantly present on the CN–HA path. In order to redirect CN's traffic intended for MN to a malicious node, the intruder most likely has to get control of a router or a switch along the CN–HA path. Furthermore, after taking over the session from MN, if the malicious node wants to continue the session with CN while pretending to be MN, the malicious node and the router need to collaborate throughout the session. For example, the router tunnels CN's traffic to the malicious node and vice versa. In the case of MIPv6, the effort committed to break the RR protocol to launch a session hijacking attack could be considerably less. Assume that MN_1 and CN are having an on-going communication session and the intruder wants to redirect CN's traffic to his collaborator MN_2. The intruder monitors the CN–HA path (i.e., anywhere from MN_1's home network to CN's network) to obtain HoT, extracts the home token CH and sends it MN_2. Upon receiving CH, MN_2 sends a CoTI to CN and CN will reply with a care-of token CC. MN_2 simply hashes the two cookies to obtain a valid session key, and uses the key to send a binding update message to CN on behalf of MN_1. The binding update will be accepted by CN which will in turn direct its traffic to MN_2.

Next, consider the malicious mobile node flooding attack. In the base IPv6 without mobility, perhaps the best example of flooding attack is the DDoS attack in which a multitude of compromised systems attack a single target. There are many ways to launch a malicious mobile node flooding attack against a victim (which can be either a node or a network) in MIPv6. For example, the malicious node starts some traffic intensive sessions with correspondent nodes and moves to the victim's network or the border between the victim network and the outside world. It then runs the RR protocol to redirect traffic from the correspondent nodes to the victim's network by sending them binding update messages. The malicious mobile node does not need any special software or networking skill to launch this attack. Due to the stateless nature of the RR protocol, it is easy for an intruder to cause havoc to, say B2C operations. Imagine a correspondent node provides on-line services to many mobile clients. The intruder can simply eavesdrop on the RR protocol messages to collect cookies on the border between the correspondent node and the Internet. The intruder then randomly hashes pairs of cookies to form session keys, and sends binding update messages to the correspondent node. This will cause redirection of traffic to randomly selected mobile clients and eventually bring down the services of the correspondent node. Hence, the RR protocol is vulnerable to many attacks. The Early Binding Updates protocol (EBU) [33] was later proposed to increase the protocol execution latency of the RR protocol by sending part of messages before an imminent handover; however, both protocols are of the same security strength, that is, the attacks applicable above are also applicable to EBU.

3.1.2. The CGA-based protocol and its analysis

The Cryptographic Generated Address (CGA) [5,6] technique for secure Binding Update, was first proposed in [29] and then in [28,14]. A 128-bit IPv6 address is divided into a 64-bit subnet prefix and a 64-bit interface identifier. "CGA technique generates a valid IPv6 address where the interface identifier is computed via a cryptographic one-way hash function from a public key and auxiliary parameters. The binding between the public key and the address can be verified by re-computing the hash value and by comparing the hash with the interface identifier" [37]. Each CGA is associated with a security parameter (Sec) and a CGA Parameters data structure (CGAP). Sec is a 3-bit unsigned integer and determines the security strength against brute-force attacks. And CGAP = (128-bit Modifier, 64-bit Subnet Prefix, 8-bit Collision Count, variable length public key, optional variable length Extension Fields). The process of generating a new CGA takes three input values: a subnet prefix, the public key of the address and the Sec. The output of the address generation algorithm is a new CGA and a CGAP. CGA verification takes as input an IPv6 address and a CGAP. It is important to note that because CGAs themselves are not certified, an attacker can create a new CGA from any subnet prefix and its own (or anyone else's) public key.

CAM-DH was proposed based on CGA technique to secure Binding Updates in [28]. In CAM-DH, each MN has a public and private key pair P_{MN} and S_{MN} in a digital signature scheme. MN's HoA is a CGA generated from P_{MN}. The protocol is described in Fig. 3, which contains five messages in total:

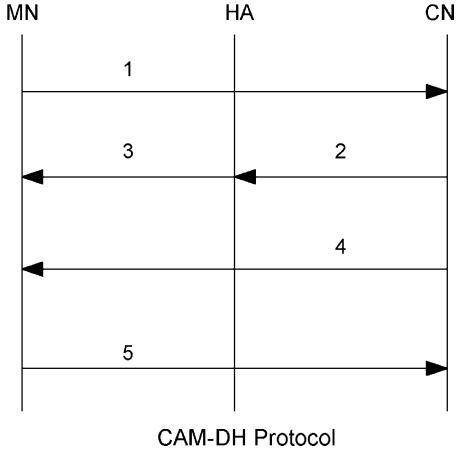


Fig. 3. The CAM-DH Protocol.

Message 1: $MN \rightarrow CN{:}HoA, CoA$.

Message 2: $CN \rightarrow MN's\ HoA{:}\{r_h, j, g^y\}$, where $r_h = MAC_{K_{CN}}(HoA|N_j|0)$.

Message 3: $HA \rightarrow MN's\ CoA{:}\{r_h, j, g^y\}$, where $r_h = MAC_{K_{CN}}(HoA|N_j|0)$.

Message 4: $CN \rightarrow MN's\ CoA{:}\{r_c, j\}$, where $r_c = MAC_{K_{CN}}(CoA|N_j|1)$.

Message 5: $MN \rightarrow CN : \{T_0, HoA, CoA, i, MAC_{K_{BU}}(T_0|HoA|CoA|i), g^x, S_{MN}(TypeTag|g^x|HoA), P_{MN}, MAC_{K_3}(T_0|\ldots P_{MN}), j\}$, where $K_3 = h(r_h|r_c)$, $K_h = h(g^{xy}|r_h)$, $K_{BU} = h(K_h|r_c)$.

In Message 1, the MN informs the CN that it is mobile, and gives both the mobile's home address (HoA) and its care-of address (CoA). In Message 2, CN sends the MN one challenge r_h to the home address, the Diffie–Hellman exponent g^y and a serial number j that indicates which version of N_j was used to generate the challenge. In Message 3, HA forwards Message 2 to MN' CoA. In Message 4, CN sends the MN one challenge r_c to the care-of address and the same serial number j as that in Message 2. In Message 5, MN sends CN a binding update, a message authentication code on the binding update computed using K_{BU}, the MN's Diffie–Hellman exponent signed with its private signature key S_{MN}, the MN's public signature key P_{MN}, and a message authentication code on all of the aforementioned data, computed using a key derived from the two challenges. This message contains a tag T_0 so that it can be distinguished from message 1 of the variant version of the protocol. The binding update also contains a sequence number i so that if more than one binding update is sent within the lifetime of a single value of N_j, it is possible to determine their relative ordering. "The *TypeTag* is used to prevent accidental type collision with messages of other protocols that use the same signature public key but may or may not use type tags" [37]. CN verifies the two MACs and can check integrity and identity of g^x by CGA.

Although the use of CGA offers CAM-DH a method for binding MN's P_{MN} to its IPv6 address, CAM-DH suffers from some limitation and problems. First, CAM-DH does not authenticate the care-of address. An attacker who can intercept packets sent to the care-of address can complete the protocol and cause the care-of address to be flooded with data, even if the host that actually owns the care-of address is not willing to participate in the protocol. An alternative method of authenticating the care-of address would have been to derive

the care-of address (as well as the home address) from the node's public key. But this approach may not be applicable, because some subnetworks may impose constraints on the care-of addresses that can be used. Second, CGA generation is computationally expensive, especially when MN uses a high Sec value. CAM-DH protocol needs much processing power because it involves asymmetric cryptography and CGA. Further, the computation load on MN side is heavy since every BU message requires the MN to generate a signature and the CN to perform a signature verification.

Another recently proposed CGA-based protocol, CGA-OMIPv6 [14], follows the same principles as in the original RR protocol and combines CGA technique and other mechanisms. Although this protocol reduces the signaling load and the handoff delay, it is exposed to the man-in-the-middle attack. Also, it is computationally expensive when generating CGA.

3.2. The CBU protocol and its analysis

In [10], we proposed the certificate-based Binding Update protocol (CBU) to solve the security problems discussed above. In the CBU protocol, the digital signature cryptosystem is also used. The public/private key pair of HA is denoted by P_{HA} and S_{HA}. The private key S_{HA} is kept by HA in the home link, probably inside a tamper-resistant hardware of cryptogram processing device. The home link obtains a public key certificate:

$$Cert_{HA} = \{HLSP, P_{HA}, Valid\_Interval, SIG_{CA}\}$$

from a Certification Authority (CA), where *HLSP* is the home link subnet prefix, *Valid_Interval* is the valid duration of the certificate, and SIG_{CA} is CA's signature on *HLSP*, P_{HA} and *Valid_Interval*. Fig. 4 shows message exchanges between a MN, its HA and its CN in CBU protocol. The existence of operations performed by HA is made transparent to CN.

When MN wants to start RO operation with CN, it sends a RO request $REQ = \{HoA, CN, n_0\}$ to CN via reserved tunnelling, where n_0 is a nonce value used to match the reply message REP. CN is used to represent both the correspondent node and its IP address. Message *REQ* is sent to MN's HA via the IPsec protected secure tunnel. Upon arriving at HA, REQ is intercepted by HA using IPv6 Neighbor Discovery [16,18]. HA will not forward REQ to CN, instead, it creates a cookie C_0 and

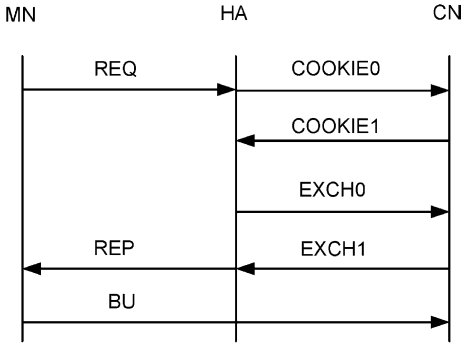


Fig. 4. The CBU Protocol.

sends $COOKIE0 = \{HoA, CN, C_0\}$ to CN. In reply, CN creates a nonce n_1 and a cookie C_1, and sends $COOKIE1 = \{CN, HoA, C_0, C_1, n_1\}$ to MN. Note that the destination address in *COOKIE1* is MN's HoA.

After receiving *COOKIE1*, HA checks on the validity of C_0, generates a nonce n_2 and a Diffie–Hellman (DH) secret value x, and computes the public value g^x. Then HA generates a signature $SIG_{HA} = S_{HA}(HoA|CN|g^x|n_1|n_2|TS)$ using its private key S_{HA}. Finally, HA replies CN with $EXCH0 = \{HoA, CN, C_0, C_1, n_1, n_2, g^x, TS, SIG_{HA}, Cert_{HA}\}$, where $Cert_{HA} = \{HLSP, P_{HA}, Valid\_Inter\ val, SIG_{CA}\}$ is the public key certificate of HA. When CN receives *EXCH0*, it validates the cookies, the HA's public key certificate $Cert_{HA}$, the signature and importantly, checks for equality of the HA's subnet prefix strings embedded in both $Cert_{HA}$ and HoA. If all the validations and checking are positive, CN can be confident that the home address HoA of MN is authorized by its HA and the DH public value g^x is freshly generated by MN's HA. CN next generates its own secret value y and its public value g^y, and then computes the DH key $K_{DH} = (g^x)^y$, a session key $K_{BU} = prf(K_{DH}, n_1|n_2)$ and a MAC $MAC1 = prf(K_{BU}, g^y|EXCH0)$, and sends $EXCH1 = \{CN, HoA, C_0, C_1, g^y, MAC1\}$ to MN. Again, this message is intercepted by HA, which first validates the cookies, calculates the $K_{DH} = (g^y)^x$ and $K_{BU} = prf(K_{DH}, n_1|n_2)$. HA then computes $MAC2 = prf(K_{BU}, EXCH1)$, and sends an optional $CONFIRM = \{HoA, CN, MAC2\}$ to CN. The validity of *MAC2* is checked by CN and if valid, CN creates a cache entry for HoA and the session key K_{BU}, which will be used for authenticating binding update messages from MN. Upon positive verification of *MAC1*, HA

also sends $REP = \{CN, HoA, n_0, K_{BU}\}$ to MN through the secure IPsec ESP protected tunnel. After receiving REP, MN checks that n_0 is the same as the one it sent out in REQ. If so, MN proceeds to send CN binding update messages protected using K_{BU} as in the RR protocol.

The protocol came up with a novel idea that creates a certificate for a home link, i.e., the subnet prefix instead of individual IP address. The argument goes as follows: First, IP addresses are often obtained by DNS (Directory Name Service) lookup and DNS does not provide a secure way of mapping names to IP addresses. Second, IP addresses are subject to renumbering both when service providers change and when configurations change so they may not be as persistent as other subject names (e.g., domain names) [25]. Third, IP addresses are leased to an interface for a fixed length of time. When an IP address's lease time expires, the association of the address with the interface becomes invalid and the address may be reassigned to another interface elsewhere in the Internet. And there might be various reasons for keeping IP addresses' lease time short, such as for privacy protection. Therefore, it is very difficult in practice for CAs to keep track of correct associations between IP addresses and all devices' interfaces in a consistent and timely manner, not to mention certificate renewal and revocation task. Subnet prefixes for home links, however, are more trackable and manageable. First, a home link subnet prefix is normally much more persistent than MN's home address. Second, the number of home links is significantly smaller than that of MNs. Third, keeping track of subnet prefixes is much easier than that of individual IP address.

The CBU protocol successfully reduces the impossible task of authenticating MN and its HoA via individual certificate to a much easier one as above mentioned. However, the protocol does not address certificate management issues for HAs. The CBU protocol is based on the assumption of the existence of fragmented authentication infrastructures within individual administration domains or even cross different domains. A CA is responsible to issue the certificate to every home link subnet prefix. But simply issuing certificate directly to every individual home link subnet prefix from one CA is still far from practical. Obviously, this flat structure of trust management is not flexible and scalable and even impossible in crossed domains resided in different administrative areas, not to mention in the global-wide range. It is also impossible for consistent and timely certificate management on renewal and revocation. Further, neither should it be CA's duty to keep tracking of the highly complicated individual subnet prefix changes of home links, nor is it possible. Note that the number of individual home links can be as high as 2^{61} and they can be anywhere around the world. Instead, a divide-and-conquer approach should be adopted to reduce the complexity and therefore, handle the flexibility and scalability problem as we will explain in the following section. The second problem with the CBU protocol is that there's no way for CN to assure MN's liveness on its claimed CoA in the BU message. This is dangerous as we have pointed out in Section 2. A spoofed BU lying on MN's CoA can be sent by any malicious node (acts as MN) and therefore, results in serious attacks.

4. Our secure routing optimization protocol: HCBU

Based on the security analysis against all above protocols mentioned so far, in this paper we propose a Hierarchical Certificate-based Binding Update protocol (HCBU). We first discuss the trust management framework of HCBU and then give the protocol details.

4.1. Trust management in HCBU

One important design rationale behind HCBU is *trust delegation*. We first analyze the role of HA in the RR protocol. In the RR protocol, HA only relays the message sent by both MN and CN. The underlying assumption is that MN has registered its CoA to HA, so that HA is aware of MN's current CoA, although the registered CoA may be a false one. Note that there's no way for HA to map MN's HoA to its real CoA if MN lies on its CoA. In the CBU protocol, the functionality of the HA is extended. HA is responsible for proving the correctness of the binding of MN and its HoA to CN. This task in the CBU protocol is achieved by issuing certificate to individual home link subnet prefix and therefore, allowing HA to sign on the message. This is reasonable because HA shares a secret key with MN and can easily authenticate MN via IPsec. Also once with a certificate in hand, HA can easily prove MN's ownership of HoA to CN via a signature. Hence in CBU, HA's duty is to assure CN: (1) MN's ownership of its HoA and (2) the BU request is indeed sent by MN. In HCBU, HA's duty is further extended to cover all the nodes

currently under its link, including roaming mobile nodes. Therefore, it will also certify a roaming MN's CoA, as long as the roaming MN's CoA is assigned by itself. HA signs on the binding of (HoA, CoA), which proves MN's ownership of CoA.

Trust Setting: Now we focus on the certificate problem and see how a global PKI can be avoided via combining a hierarchical certificate structure with the inherent unicast address structure of IPv6. A 3-layer hierarchical trust management framework is adopted in HCBU to achieve flexibility and scalability as shown in Fig. 5. Basically, this is a divide-and-conquer approach, taking advantage of the underlying IPv6 address assigning and allocation mechanism. In the 1st layer, based on the existing trust relationships among Tier-1 ISPs, one or several mutually agreed CAs are chosen to issue a special certificate to each ISP. And each certificate at the minimum contains the following contents: (a) TLAs owned by the given ISP; (b) public key of the ISP; (c) valid interval; (d) a CA's signature on (a), (b) and (c). In the 2nd layer, each Tier-1 ISP issues certificate from its own domain to Next Level Aggregators (NLAs) of its downstream intermediate ISPs. The certificate structure is the same with the former one with slight difference in contents: (a) NLAs owned by the intermediate ISP; (b) public key of that intermediate ISP; (c) valid interval; (d) Tier-1 ISP's signature on (a), (b) and (c) using its own private key. This certificate issuing operation could be repeated several times according to the real situations. Each upper level ISP issues certificates to its immediate down stream ISPs from its own domain. Generally, the length of this certificate chain would not be long in practice. The certificate's valid interval at each lower layer should be shorter than that of the upper layer certificate. Finally, each home link gets a certificate of its own on its Site Level Aggregator (SLA). At this point, all the routing information (i.e., TLA + NLA + SLA) in the subnet prefix(s) of a home link has been approved by the certificates from both the 1st and 2nd layer. Then in the 3rd layer, each home agent dynamically signs on the binding of (MN, HoA) or (HoA, CoA) upon request, which proves to CN the correctness of the binding. (We will discuss it in detail later.)

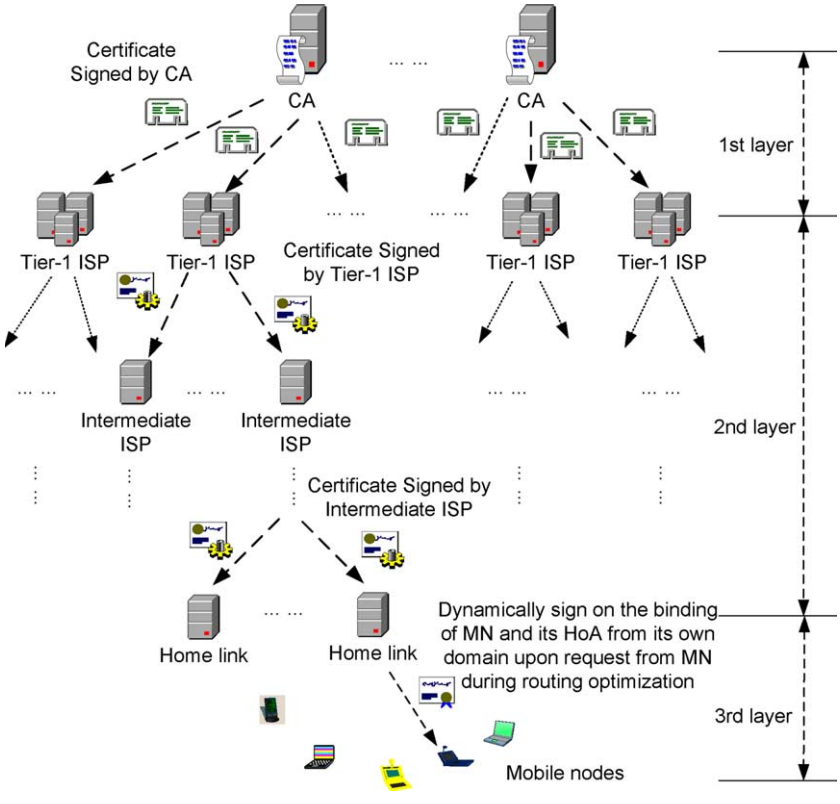


Fig. 5. The 3-layer trust management framework.

Trust Delegation: In our trust management framework, the home agent of MN's roaming link signs the binding of (HoA, CoA). Note that this signature only proves MN's ownership of CoA and is only used for this purpose. It does not imply MN's ownership of HoA. This signature together with a valid subnet prefix certificate chain of that link convinces MN's HA that MN actually owns this CoA after receiving this information in the care-of address registration sent by MN. Subsequently, upon successful verifying MN's ownership of its claimed CoA, HA now proves to CN the correctness of the binding of (HoA, CoA) by signing it through its own private key. In turn, CN is convinced that MN actually owns both HoA and CoA by verifying HA's signature and its subnet prefix certificate chain. Therefore, MN proves to CN its ownership of both its HoA and CoA by delegating the job to its own HA. At the same time, MN is also exempted from computation-expensive signature operations due to the delegation. We note that this delegation significantly improves the performance of our routing optimization protocol as we will see in the next section.

As a consequence of our framework, during the RO process, a signed message together with a certificate chain ($Cert\_Chain_{HA}$) is sent from HA to CN upon request from MN, which proves MN's ownership of both its HoA and CoA. The $Cert\_Chain_{HA}$ is a series of certificates with the first certificate issued by CA and the last issued by HA's immediate upper stream ISP. A sample home link certificate a HA may hold can be expressed as below at the minimum.

$$\{SLA, P_{HA}, Valid\_Interval, SIG_{ISP}\}$$

The intermediate certificates in the *Cert_Chain* are issued by intermediate ISPs. So it is clear that in our trust management framework, the subnet prefix of a MN is proved by the certificate chain, and the interface identifier is proved by the HA signed message. At the CN side, certificates of 1st and 2nd layers can be easily checked by caching a small set of frequently used ones. Generally, CN only needs to actually verify the signature(s) in the last one or two certificates at the end of the certificate chain. More over, should CN be a mobile node itself, the checking task can be delegated to its own HA easily.

Remarks. In our framework, CA is required to issue certificates only to Tier-1 ISPs. The argument goes as follows: (1) Tier-1 ISPs have steady network prefixes (i.e., TLAs) and a very small number. Therefore, they can be reasonably tracked by CA. (2) The existence of well established long term trust relationships and extensive cooperations among these Tier-1 ISPs makes it easy to find mutually agreed CA(s) to issue the certificate of such a kind. (3) The Tier-1 ISPs cover most parts of the Internet. (4) Due to the hierarchical structure of IPv6 unicast address itself, address changes at lower layer are transparent to the upper layer and thus have no impact on the upper layers' certificates.

Considering the difficulty of the deployment issue, the certificate verification between CN and an arbitrary 3rd layer agent may not be trivial. However, this problem can be eased by combining the use of our subnet prefix certificates with those for SEND [2]. The gradual build-up of SEND infrastructure could ease transition to HCBU. In addition, HCBU may also obtain help from the AAA infrastructure [13].

4.2. The HCBU protocol

As in both RR and CBU protocols, all the protocol messages in HCBU are carried within IPv6 "Mobility Header". The protocol messages exchanged among a MN, its HA and CN are shown in Fig. 6. Before a secure BU assured by CN can be sent, the following three steps are followed by the three protocol participants. In HCBU, when a MN is roaming to a foreign link, it obtains a CoA as usual and additionally, MN requests a signature on the binding of (HoA, CoA): $SIG_{HA'} = S_{HA'}(HoA, CoA, Valid\_Interval)$.

In the 1st step, MN first initializes a Binding Update request in Message 1, when it realizes an imminent handover:

Message 1. Binding Update Request (BUReq):

$$\{BU, N_m, HoA, CN\}.$$

Message 1 contains MN's own home address, a fresh random nonce, and CN's address in addition to Binding update preparation request (BU). (We use CN to denote both the node and its corresponding IP address.) Message 1 is sent to HA through the pre-established secure tunnel via IPsec. CN's address is included in the message to clearly indicate the destination of the BU request. HA next does pre-exchanges with CN to prepare the coming binding update through Message 2 and Message 3.

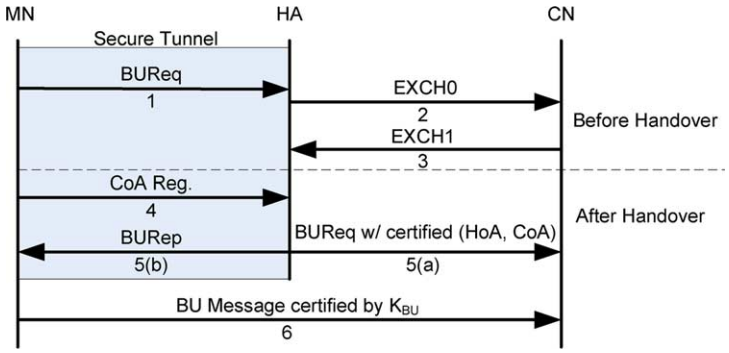


Fig. 6. The HCBU Protocol.

Message 2. Pre-Information Exchange0 (EXCH0):

$\{N_m, HoA, CN, g^x\}$.

Message 3. Pre-Information Exchange1 (EXCH1):

$\{N_m, N_c, HoA, CN, g^x, g^y, Cookie_{CN}\}$,

where

$Cookie_{CN} = prf(K_{CN}, N_m|N_c|HoA|CN|g^x|g^y)$.

Message 2 passes the fresh nonce N_m, MN's HoA, CN's address and a DH public value g^x to CN. In reply, CN attaches its own fresh nonce N_C and DH public value g^y to the received Message 2 and thus forms Message 3. CN next creates a cookie $Cookie_{CN}$ for HA using its own secret key K_{CN}. We will discuss more on the two DH public values later. At this point, CN does not create a state for the protocol. This is useful for protecting CN from resource exhausting attack.

In the 2nd step, MN first proves to HA its ownership of CoA. HA then proves to CN MN's ownership of both its CoA and HoA. A session key is also established between HA (on behalf of MN) and CN to certify the final BU message between MN and CN during the 3rd step. The DH key exchange method is used. The 2nd step consists of 3 messages.

Message 4. Care-of address Registration:

$\{CoA, HoA, Valid\_Interval, CN, SIG_{HA'}, Cert\_Chain_{HA'}\}$.

HA checks the validity of the certificate chain and verifies the signature contained in the message. Negative result of either of them leads to the rejection of the message. Note that this checking operation assures the correctness of MN's CoA. Message 4 actually can be a piggy-backed part of MN's care-of address registration message. Note that when the mobile node moves to a different network, and is configured a new care-of address, the mobile node must first register the new care-of address with its home agent together with other operations, before it can use the new care-of address [12].

Next Message 5 completes the preparation for the Binding Update. Message 5(a) passes all the required information for Routing optimization to CN, including the information HA obtains in Message 1, the cookie obtained in Message 3 and HA's signature on these information. Finally, HA's certificate chain is also attached.

Message 5(a). Binding Update Request with Certified (HoA, CoA):

$\{N_m, g^x, Cookie_{CN}, HoA, CoA, Valid\_Interval, CN, SIG_{HA}, Cert\_Chain_{HA}\}$,

where

$SIG_{HA} = S_{HA}(HoA|CoA|Valid\_Interval|CN|g^{xy}|N_m|N_c)$

On arriving at CN, Message 5(a) is processed in the following sequence: (1) Validate the cookie $Cookie_{CN}$; (2) Check on the authenticity of the certificate chain $Cert\_Chain_{HA}$; (3) Calculate DH key K_{HS}, verify the signature, and importantly, check for the equality of the home link subnet prefix strings embedded in both $Cert\_Chain_{HA}$ and HoA. The included fresh nonce assures the freshness of the signature. If all the results of validation and checking operations are positive, CN now can be confident that both MN's HoA and CoA are indeed correct. At this point, CN creates a cache for (HoA, CoA) and the Binding Update key $K_{BU} = prf(g^{xy}, N_m|N_c)$. Now CN only needs to wait a final message (Message 6) from MN to make sure MN is still alive on its CoA. At the same time, MN obtains the Binding Update key K_{BU} in Message 5(b) from HA and therefore, could send out the final Binding

Update message. Note that Message 5(b) is actually sent before Message 5(a) is sent so that Message 6 can be sent earlier.

Message 5(b). Binding Update Reply (BURep):

$\{HoA, CoA, CN, K_{BU}\}$.

Note that Message 5(b) is sent through IPsec as well and this completes the 2nd step. Upon getting K_{BU}, MN now sends out the final message:

Message 6. Binding Update Message certified by K_{BU}:

$\{HoA, CN, CoA, MAC\}$,

where

$MAC = prf(K_{BU}, N_m|HoA|CN|CoA)$.

Upon receiving Message 6, CN easily verifies that MN is still alive on its CoA. In case that Message 6 arrives at CN before Message 5(a), CN will simply discard it. Considering that buffering Message 6 at CN may cause flooding attacks, we require MN to resend Message 6, if MN still receives data sent by CN from HA. Although this setting may sacrifice the performance a little bit (because of the delay between Message 5(a) and the next Message 6), it helps HCBU be highly robust against such flooding attacks. Thus, this last step completes our routing optimization protocol.

Note that K_{BU} is the shared secret key to certify the binding update between MN and CN and therefore, K_{BU} proves MN's ownership of HoA. By making use of K_{BU}, the subsequent Binding Updates between MN and CN can be much more efficient as shown below:

Binding Update Message certified by K_{BU}:

$\{HoA, CoA, SIG_{HA'}, N'_m, CN, Cert\_Chain_{HA'}, MAC\}$,

where

$MAC = prf(K_{BU}, N'_m|HoA|CN|CoA)$.

Upon receiving this binding update message, CN first checks the integrity of the attached MAC, and thus verifies the message is indeed sent by MN. Next, CN checks HA''s certificate and verifies $SIG_{HA'}$. If both verifications succeed, CN now assures that MN is indeed alive in the CoA as claimed in the previous messages. Further, both MN and CN update $K_{BU} = prf(K_{BU}, N'_m)$ in order to prevent replay attack by resending the same message. Note that only one message is needed in this case to accomplish routing optimization between MN and CN. The HCBU protocol treats this scenario as a special case of the above general protocol.

4.3. Analysis of the HCBU protocol

The HCBU protocol achieves high security strength by adopting certificate-based strong authentication approach, and keeps low computation costs by relying on the underlying trust management framework. The communication efficiency of HCBU protocol is achieved by integrating the technique introduced in the EBU protocol [33].

4.3.1. Security analysis

In Message 1, BUReq is sent from MN to its HA through the pre-established secure tunnel via IPsec; hence, only a legal mobile node belonging to that particular home link can initialize this message. The following messages exchanged between HA and CN provide strong one-way authentication to CN on MN's ownership of its HoA and CoA, because the signature on subnet prefix and the certificate chain of HA are used. The design technique of these messages is basically adapted from well-known JFKr protocol [1]. In Message 2 and Message 3, by pre-computing and reusing the DH public value g^x and g^y, neither HA nor CN commits much processing resources, except for the fresh nonce and cookie generating. This prevents DoS attack against both HA and CN.

After receiving Message 5(a), CN checks on the equality of the home link subnet prefix contained in both certificate chain and HoA. The checking is critical to detect man-in-the-middle attack. The signature in the message serves for three purposes: First, it certifies that the DH value g^x was originated by MN's home agent HA on behalf of MN. Second, it testifies that HoA is under HA's (or equivalently the home link's) jurisdiction and is a legitimate home address for its mobile node MN, which authenticates MN's HoA to CN. Third, because MN's HA signs on the binding of (MN, HoA, CoA), implying that HA has already verified the authenticity and valid interval of MN's CoA, CN is therefore exempted from the computation-expensive checking operations on the correctness of MN's CoA. Hence, by verifying HA's signature SIG_{HA}, CN now assures the authenticity of both MN's HoA and CoA. Finally, CN assures MN's liveness on its CoA by getting Message 6.

Since a successful completion of the protocol allows CN to authenticate MN's HoA as well as

allows the two nodes to set up a secret session key for certifying the Binding Updates, the protocol prevents any spoofed BU message attack. In addition, by introducing our trust management framework, our protocol assures MN's CoA to CN. Lacking of assurance on MN's CoA results in attacks on all the other protocols based on Routing Returnability technique, because a malicious node can lie on its CoA while uses its authenticated HoA. As discussed in the RR protocol, an attacker only needs to eavesdrop the exchanged messages, and easily mounts attacks by convincing CN to accept spoofed BUs. In the CBU protocol, CN never assures MN's liveness on its claimed CoA; a malicious node can claim any CoA arbitrarily in the BU Message, and there's no requirement of the ability to present itself on the route between CN and this claimed CoA. In our protocol, CN assures the authenticity of both MN's HoA and CoA by obtaining the signed message from MN's HA, and MN's liveness by getting the binding update from MN certified by a fresh shared secret key. Therefore, such malicious node attack is not valid in HCBU.

Another point needs to be addressed is the authentication of HA's certificate chain at CN's side. In MIPv6, CN can be any node ranging from a powerful web server, a regular static node to a much weaker mobile node. When CN is a web server, it is easy for CN to cache the 1st layer and 2nd layer certificates; hence there should be no problem for CN to authenticate HA's certificate. If CN is a regular static node, CN can cache a small set of frequently used upper level certificates, and seek help from its own home link whenever needed. In case that CN is a mobile node itself, CN can delegate the certificate authentication job to its own home agent. In that case, the authentication messages will be exchanged between the two HAs, and then the results are tunnelled to the respective mobile node. The potential inter-domain authentication problem as discussed in Section 4.1 can be eased with the help of SEND certificates and the AAA infrastructure [2,13].

4.3.2. Computation and communication efficiency analysis

Computation Efficiency Analysis: HCBU tries to minimize the number of computation-expensive operations, and especially, MN is exempted from such operations by design. In Message 2 and Message 3, by introducing *Forward Secrecy Interval* technique, the DH public value g^x and g^y can be reused in different protocol runs by HA and CN, respectively. The freshness of the established DH key is guaranteed by the two fresh nonces N_m and N_C. The detailed discussion on this issue can be found in [1]. By doing so, two goals are achieved: First, the computation burdens on both sides are significantly reduced. Both g^x and g^y only need to be refreshed periodically. To achieve the same goal, HA and CN in the CBU protocol must compute a fresh DH public value in every protocol run, respectively. Note that Binding Update in MIPv6 happens extensively. Therefore, it is critical to reduce the computation burden on both HA and CN sides.

Hence, in HCBU the computation-extensive operations of each protocol participant are as follows: (1) MN: none; (2) HA: one signature verification, one certificate verification, one signature operation, one DH value generation operation per Forward Secrecy Interval and one DH key calculation operation; (3) CN: one certificate verification, one signature verification, one DH value generation operation per Forward Secrecy Interval and one key calculation operation. Note that in CAM-DH [5], MN is required to do: one DH value generation operation and one DH key calculation operation in addition to one signature operation, while at CN side, the computation costs are mostly as same as that in HCBU: only one DH value generation operation per Forward Secrecy Interval is required additionally.

HA's computation costs are further significantly reduced as HCBU does not require HA's participation when CN is known to MN as we discussed in Section 4.2. The computation costs at CN side can be further reduced, if CN delegates its verification task to its own home link when CN itself is a mobile node.

Communication Efficiency Analysis: In HCBU, to complete the whole routing optimization operation, MN is required to send 3 messages, HA is required to send 3 messages and CN sends only one message. In total, 7 messages are needed for the three protocol participants. Also note that Message 4 is actually combined with the care-of address registration message and thus is not an additionally generated message. Moreover, if MN has multiple concurrent sessions ongoing with different or the same CN(s), the proposed HCBU could be more economical. In this scenario, MN sends only one Message 1 to HA with a list of CN addresses, instead of a separate message for each CN and each session. Note that the same suppressing technique can be applied to

Message 4 and 5(b). This further improves communication efficiency of the proposed HCBU protocol.

The protocol latency is significantly reduced by executing the 1st step (Messages 1–3) right before an imminent handover. If handovers cannot be anticipated, the mobile node may periodically repeat the 1st step, so that they can always be prepared for an unexpected handover. By executing the 1st step before the handover, HA and CN are thus able to compute K_{BU} beforehand, which further saves processing time. Hence, right after the handover happened, MN registers its CoA to HA and proves its ownership of CoA at the same time by sending Message 4 along with the registration message. And HA is quickly able to send out Messages 5 with K_{BU} already in hand. Therefore, the protocol latency is fairly low by pre-sending messages before the handover and controlling the number of computation-expensive operations after the handover. Vogt et al. [33] claims that this technique would reduce the protocol latency to half and even more as compared to the RR protocol. Note that lower latency and higher efficiency may be critical for a mobile node to successfully finish the protocol in the error-prone wireless communication environment.

4.4. Support for hierarchical MIPv6

In the protocol of Hierarchical Mobile IPv6 (HMIPv6) [30], the concept of Mobility Anchor Point (MAP) is introduced. MAP is a router located in a domain visited by the mobile nodes. MAP provides the localized mobility management for the visiting mobile nodes. Every mobile node bundles three addresses: Home Address (HoA), Regional Care-of Address (RCoA), and On-Link Care-of Address (LCoA). RCoA is an address on the MAP subnet, and obtained by the mobile node (MN) from the visited domain. LCoA is configured on a MN's interface based on the prefix advertised by its default router. In fact, it is a care-of address in normal MIPv6. Fig. 7 shows the architecture of HMIPv6. Note that the hierarchy in HMIPv6 can be arbitrarily deep.

In HMIPv6, when CN sends packets to MN's RCoA, MAP intercepts the packets and forwards the packets to MN's LCoA. However, as the BU message from MN to MAP is not authenticated when MN changes its Access Router (AR), attackers can easily launch attacks, which redirect the traffic from MAP to fake destinations chosen by the attackers. An earlier solution appeared in [24] is based on the extension of the CBU protocol, however it suffers from many security flaws.

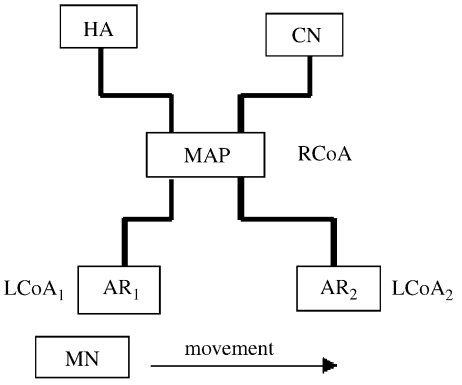


Fig. 7. Hierarchy MIPv6 architecture.

In this subsection, we propose our new approach to protect the BU message from MN to MAP. When MN roams in the MAP domain, as soon as MN attaches to another AR and gets a new LCoA, MN will send a BU message to MAP

$$BU = \{T, LCoA, MAP, HoA, RCoA, SIG_{MN}, Cert_{MN}, Cert\_Chain_{HA}\},$$

where

$$SIG_{MN} = S_{MN}(T|LCoA|MAP|HoA|RCoA),$$

and

$$Cert_{MN} = \{HoA, P_{MN}, Valid\_Interval, MAP\_list, SIG_{HA}\}.$$

S_{MN} and P_{MN} denote the private and public key pair of MN. In order to support localized micro-mobility management feature in HMIPv6, a special public key certificate of MN, issued by its HA, is designed as above. Here, SIG_{HA} is HA's signature on a time stamp T, MN's HoA, P_{MN}, MAP_list and *Valid_Interval*. The time stamp is enclosed to prevent replay attack. Once the MN enters a foreign link which supports HMIPv6, it can then request its HA to issue such a certificate for him, which is used only for proving the correctness of the binding of MN and its HoA to a small set of MAPs. The MAP addresses are included in the certificate as *MAP_list* to confine the applicable range of the certificate. The *Valid_Interval* is set appropriately according to MN's alive time on the foreign link. Generally, the certificate should expire quickly. Therefore, every time when MN enters a new foreign link out side the certified MAP domains, it

must first get such a certificate for subsequent BUs. Then MN will send the signed BU message using the public key P_{MN} provided by the certificate, whenever it attaches to another AR and gets a new LCoA. The certificate chain of HA is attached in the BU messages to facilitate the authentication process by MAP. Note that through this design, all the BU related traffic is the local traffic (i.e., within the MAP domain), except for occasionally one communication round between MN and its HA, which is related to the certificate request and issuing.

By double-checking the equality of home link's subnet prefix string embedded in both $Cert\_Chain_{HA}$ and HoA, MAP can finally trust MN's new BU message. Still the authentication itself cannot completely prevent the malicious node from lying on its LCoA in the BU message and thus mounting an attack. In our design, two countermeasures are used to frustrate the attack: (1) Different MAP domain outside the certified ones requires different certificate so that the attacker could not arbitrarily choose the target link before getting the according certificate. In [24], the certificate is held by MN all the time and is applicable at every link, while in our approach such certificate is issued dynamically upon request from MN to the specified MAP domain. Obviously, under their approach a malicious MN can always mount the attack at any link once with a certificate in hand. (2) In our approach, MN's certificate is deliberately designed to expire quickly. Any malicious node that is detected to mount attacks will be forbidden to renew its certificate. This reduces the attack damage caused by a MN to the minimum. The techniques and solutions of intrusion detection in wireless LANs can be found in [38,39] and are out of the scope of this paper. Also note that the protocol in [24] suffers from replay attack, because the old BU messages can be replayed by the attackers at any time. Therefore, The attacker can successfully redirect the traffic destined to MN's current LCoA to its old LCoA. In our protocol, the time stamp T is included both in the first BU message and the signature and therefore frustrate the replay attack. Note that this time stamp is only of local significance, that is, it is valid only within the MAP subnet.

5. Concluding remarks

The fast Internet evolution together with the enormous growth in the number of users of wireless technologies has resulted in a strong convergence trend towards the usage of IP as the common network protocol for both fixed and mobile networks. Future All-IP networks will allow users to maintain service continuity while moving through different wireless systems. However, introduction of mobility into IP also brought with it new security issues and attacks, among them attacks against Routing Optimization in MIPv6 are the ones need the most serious attention.

In this paper, we first discussed the backgrounds in MIPv6 security design. The assumptions and starting point of the security designs in MIPv6 are first elaborated in detail. Next we analyzed the security threats in MIPv6, and classified the attacks against Routing Optimization into two kinds. Attacks of the first kind try to exploit the spoofed Binding Update messages to achieve attackers' purpose (e.g., redirecting the traffic). The second kind of attacks tries to attack the Binding Update protocol itself, and prevent the protocol participants from correctly completing the protocol (e.g., *Resource Exhausting*). We then reviewed three existing protocols, including the ones in current Internet draft and a certificate-based approach. The pros and cons of these protocols were discussed in detail, as well as the underlying design rationale, in order to give an in-depth understanding.

We then proposed our Hierarchical Certificate-based Binding Update protocol (HCBU). A flexible and scalable 3-layer trust management framework is first introduced in order to solve the difficult certificate issuing and management problem presented in the previous certificate-based solutions, which is a divide-and-conquer approach in consideration of the underlying IPv6 address assigning mechanism and home link's jurisdiction over the addresses it assigns. In our framework, CA is required to issue certificate to only Tier-1 ISPs, which minimizes the number of required certificates managed by CA(s). Also this layered-out structure hides the changes and complexities of the lower layer from the upper layer, fully conforms to the hierarchical structure of IPv6 address format, and therefore, is relatively simple and efficient in implementation, requiring minimum additional efforts in deployment. Based on the trust management framework established above, we came up with our binding update protocol in which both a mobile node's home address and care-of-address can be easily authenticated to CN via MN's HA and thus a much stronger security strength is achieved. The proposed protocol is efficient and light-weight in the following sense: (1)

Certificate management in our protocol is relatively simple and efficient. (2) Computation costs on the protocol participants are significantly reduced, compared to the previous certificate-based protocols; (3) The latency of our HCBU protocol is fairly low, which ensures fast handovers. The communication efficiency is achieved in our protocol by taking advantage of early binding update technique [33]. Finally, an extended protocol was also proposed to give explicit support to HMIPv6.

Acknowledgements

The authors would like to thank anonymous referees for their constructive comments on the draft, which helped improve the quality of this paper substantially.

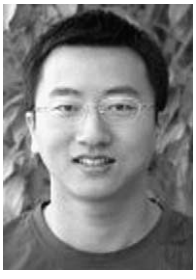
References

[1] W. Aiello et al., Efficient, DoS-resistant, secure key exchange for Internet protocols, in: Proceedings of the ACM Computer and Communications Security (CCS) Conference, Washington, 2002, pp. 48–58.
[2] J. Arkko, J. Kempf, B. Sommerfeld, B. Zill, P. Nikander, SEcure Neighbor Discovery (SEND), IETF RFC 3971, 2005.
[3] J. Arkko, Using IPsec to Protect Mobile IPv6 Signaling between Mobile Nodes and Home Agents, IETF RFC 3776, 2004.
[4] T. Aura, <http://research.microsoft.com/users/tuomaura/Publications/aura-roe-arkko-acsac02-slides.ppt>.
[5] T. Aura, Cryptographically Generated Address (CGA), IETF RFC 3972, 2005.
[6] T. Aura, Cryptographically generated addresses (CGA), in: Proceedings of the 6th Information Security Conference (ISC'03), Bristol, UK, LNCS, vol. 2851, 2003.
[7] T. Aura, Mobile IPv6 security, in: Proceedings of the Security Protocols, 10th International Workshop, Cambridge, UK, April, LNCS, vol. 2467, 2002.
[8] T. Aura, M. Roe, J. Arkko, Security of Internet location management, in: Proceedings of the 18th Annual Computer Security Applications Conference, Las Vegas, 2002, pp. 78–87.
[9] J. Bound et al., Dynamic Host Configuration Protocol for IPv6 (DHCPv6), IETF draft-ietf-dhc-dhcpv6-23.txt, 2002.
[10] R. Deng, J. Zhou, F. Bao, Defending against redirect attacks in mobile IP, in: Proceedings of the of 9th ACM Conference on Computer and Communications Security (CCS), Washington, 2002, pp. 59–67.
[11] S. Deering, R. Hinden, Internet Protocol, Version 6 (IPv6) Specifications, IETF RFC 2460, December 1998.
[12] D. Johson, C. Perkins, Mobility Support in IPv6, RFC 3775, 2004.
[13] F. Dupont, J. Bournelle, AAA for Mobile IPv6, draft-dupont-mipv6-aaa-01.txt, Expired IETF Internet Draft, November 2001.
[14] W. Haddad, L. Madour, J. Arkko, F. Dupont, Applying Cryptographically Generated Addresses to Optimize MIPv6 (CGA-OMIPv6), draft-haddad-mip6-cga-omipv6-03.txt, Expired IETF Internet draft, 2004.
[15] H. Krawczyk, M. Bellare, R. Canetti, HMAC: Keyed-Hashing for Messaging Authentication, IETF RFC 2104, 1997.
[16] A. Mankin et al., Threat Models Introduced by Mobile IPv6 and Requirements for Security in Mobile IPv6, IETF draft-ietf-mipv6-scrty-reqts-02.txt, Work in progress, 2002.
[17] MOREnet. Available from: <http://www.more.net/technical/research/ipv6>.
[18] T. Narten, E. Nordmark, W. Simpson, Neighbor Discovery for IP Version 6 (IPv6), IETF RFC 2461, December 1998.
[19] P. Nikander, T. Aura, J. Arkko, G. Montenegro, Mobile IP version 6 Route Optimization Security Design Background, Expired IETF Internet Draft, 2003.
[20] P. Nikander, T. Aura, J. Arkko, G. Montenegro, Mobile IP version 6 (MIPv6) Route optimization security design, in: Proceedings of the IEEE Vehicular Technology Conference Fall 2003, Orlando, 2003.
[21] NIST, Digital Signature Standard, NIST FIPS PUB 186, 1994.
[22] NIST, Secure Hash Standard, NIST FIPS PUB 180, 1993.
[23] C. Perkins, IP Mobility Support, IETF RFC 2002, October 1996.
[24] Y. Qiu, J. Zhou, F. Bao, Protecting All Traffic Channels in Mobile IPv6 Network, IEEE Wireless Communications and Networking Conference 2004 (WCNC 2004), Atlanta, 2004.
[25] E. Rescorla, SSL and TLS: Designing and Building Secure Systems, Addison-Wesley, 2001.
[26] R. Rivest, The MD5 Message Digest Algorithms, IETF RFC 1321, 1992.
[27] R. Rivest, A. Shamir, L. Adleman, A Method for Obtaining Digital Signatures and Public-Key Cryptosystems, Commun. ACM 21 (1978) 120–126.
[28] M. Roe, T. Aura, G. O'Shea, J. Arkko, Authentication of Mobile IPv6 Binding Updates and Acknowledgments, draft-roe-mobileip-updateauth-02.txt, Expired IETF Intenet draft, 2002.
[29] G. Shea, M. Roe, Child-Proof Authentication for MIPv6 (CAM), Computer Communications Review, April 2001.
[30] H. Soliman, K. El-Malki, Hierarchical MIPv6 Mobility Management (HMIPv6), Internet Draft, draft-ietf-mipshop-hmipv6-04.txt, Work in progress, 2004.
[31] S. Sudanthi, Mobile Ipv6. Available from: <http://www.giac.org/practical/GSEC/Sudha_Sudanthi_GSEC.pdf>, 2003.
[32] S. Thomas, T. Narten, IPv6 Stateless Address Autoconfiguration, IETF RFC 2462, 1998.
[33] C. Vogt, R. Bless, M. Doll, T. Kfner, Early Binding Updates for Mobile IPv6, Expired IETF Internet-Draft, draft-vogt-mip6-early-binging-updates, August 2004.
[34] K. Seo, C. Lynn, S. Kent, Public-Key Infrastructure for the Secure Border Gateway Protocol (S-BGP), DARPA Information Survivability Conference and Exposition paper, 2001. Available from: <http://www.net-tech.bbn.com/sbgp/sbgp-index.html>.
[35] S. Kent, C. Lynn, K. Seo, Secure border gateway protocol (Secure-BGP), IEEE J. Select. Areas Commun. (JSAC) 18 (4) (2000) 582–592.
[36] <http://www.lancs.ac.uk/postgrad/pissias/netsec/ddos/>.

[37] P. Nesser, II, A. Bergstrom (Ed.), Survey of IPv4 Addresses in Currently Deployed IETF Security Area Standards Track and Experimental Documents, IETF RFC 3792, 2004.
[38] <http://www.airdefense.net/>.
[39] J. Wright, Layer 2 Analysis of WLAN Discovery Applications for Intrusion Detection. Available from: <http://home.jwu.edu/jwright/papers/l2-wlan-ids.pdf>.

Kui Ren received his B.Eng and M.Eng both from Zhejiang University, China, in 1998 and 2001, respectively. He worked as a research assistant at Shanghai Institute of Microsystem and Information Technology, Chinese Academy of Sciences from March 2001 to January 2003, then in Institute for Infocomm Research, Singapore from January 2003 to August 2003, and in Information and Communications University, South Korea from September 2003 to June 2004. Currently he is a Ph.D. student at ECE department of Worcester Polytechnic Institute. His research interests include ad hoc/sensor network security, wireless mesh network security, Internet security, and security and privacy in ubiquitous computing environments.

Wenjing Lou is an assistant professor in the Electrical and Computer Engineering department at Worcester Polytechnic Institute. She obtained her Ph.D. degree in Electrical and Computer Engineering from University of Florida in 2003. She received the M.A.Sc degree from Nanyang Technological University, Singapore, in 1998, the M.E degree and the B.E degree in Computer Science and Engineering from Xi'an Jiaotong University, China, in 1996 and 1993 respectively. From December 1997 to July 1999, she worked as a Research Engineer in Network Technology Research Center, Nanyang Technological University. Her current research interests are in the areas of ad hoc and sensor networks, with emphases on network security and routing issues.



Kai Zeng received his BS degree in Communication Engineering and MS degree in Communication and Information System from Huazhong University of Science and Technology, China, respectively in June, 2001 and June, 2004. He is currently a Ph.D. student majored in Electrical and Computer Engineering department at Worcester Polytechnic Institute. His research interests are in wireless networking, especially on mobile wireless ad hoc networks and sensor networks with an emphasis on routing and network security.

Jianying Zhou is a lead scientist at Institute for Infocomm Research (I2R), and heads the Internet Security Lab. He is also an adjunct senior scientist in University of Malaga, adjunct associate professor in Singapore Management University and adjunct professor in University of Science and Technology of China. He obtained Ph.D. degree in Information Security from University of London (sponsored by UK government and K.C. Wong Education Foundation), MSc degree in Computer Science from Chinese Academy of Sciences, and BSc degree in Computer Science from University of Science and Technology of China. His research interests are in computer and network security, cryptographic protocol, digital signature and non-repudiation, mobile communications security, public-key infrastructure, secure electronic commerce, and virtual private network. He is actively involved in the academic community, serving on international conference committees and publishing papers at prestigious technical conferences and journals. He is a world-leading researcher on non-repudiation, and authored the book Non-repudiation in Electronic Commerce which was published by Artech House in 2001. He is a co-founder and steering committee member of International Conference on Applied Cryptography and Network Security, and served as program chair of ACNS 2003 and general chair of ACNS 2004. He received National Science and Technology Progress Award from State Commission of Science and Technology in 1995 in recognition of his achievement in the research and development of information security in China.

Feng Bao received his BS in mathematics, MS in computer science from Peking University and his Ph.D. in computer science from Gunma University in 1984, 1986 and 1996 respectively. He was an assistant/associate professor of the Institute of Software, Chinese Academy of Sciences 1987–1993 and a visiting scholar of Hamberg University, Germany 1990–1991. Since 1996 he has been with Institute of System Science, Kent Ridge Digital Labs, Labs for Infocomm Technology and Institute for Infocomm Research, Singapore. Currently he is a Principal Scientist and the Head of the Cryptography Lab of the Institute for Infocomm Research. His research areas include algorithm, authomata theory, complexity, cryptography, distributed computing, fault tolerance and information security. He has published more than 130 international journal/conference papers and owned 18 patents.

Robert Deng received his B.Eng from National University of Defense Technology, China, his M.Sc and Ph.D. from Illinois Institute of Technology, Chicago. He is currently Professor and Director of SIS Research Center, School of Information Systems, Singapore Management University. Prior to this, he was Principal Scientist and Manager of Infocomm Security Department, Institute for Infocomm Research. He has more than 20 patents and more than 140 technical publications in international conferences and journals in the areas of digital communications, network and distributed system security and information security. He has served as general chair, program chair, and program committee member of numerous international conferences. He received the University Outstanding Researcher Award from the National University of Singapore in 1999.