Improved Security Mechanism for Mobile IPv6

Jing Li, Po Zhang, and Srinivas Sampalli
(Corresponding author: Srinivas Sampalli)

Faculty of Computer Science, Dalhousie University
6050 University Ave, Halifax, Nova Scotia B3H 1W5, Canada (Email: srini@cs.dal.ca)

(Received Mar. 4, 2006; revised and accepted Apr. 29, 2006 & July 31, 2006)

Abstract

Security is a critical design issue in Mobile IPv6 since adversaries can take advantage of its routing process and arbitrarily channelize the traffic to different destinations. The original security scheme, the return routability (RR) procedure, used in Mobile IPv6 route optimization does not protect against adversaries who are on the path between the home agent (HA) and the correspondent node (CN) [11]. In addition, the long latency associated with the return routability test can impact delay-sensitive applications. This paper presents a hash chain based security mechanism to improve the security and performance of the return routability procedure in Mobile IPv6. Our fast authentication mechanism utilizes the hash chain element as an extra certificate to the mobile node in authenticating binding updates while running the routing process. In addition, by removing the necessity of the home test procedure in the RR test our mechanism can reduce the binding update latency. We analyze the security strength of our scheme under different adversary scenarios. Furthermore, a performance comparison of our scheme and the original RR procedure is provided

Keywords: Hash chain, mobile IPv6, return routability test, security

1 Introduction

MOBILE IP is an IP layer mobility protocol that is designed to support seamless roaming to mobile nodes on the Internet, without requiring modifications to current network devices and applications. Currently, there is a strong research and development interest in Mobile IPv6. Figure1 depicts the Mobile IPv6 architecture. A mobile host, called *mobile node* (MN), may have multiple IP addresses at the same time. When it is currently attached to its home network, it is only addressed by its *home address* (HoA). The HoA remains unchanged regardless of where the mobile node is attached to the Internet, unless its home subnet prefix is changed. Hence, the HoA becomes not only an address that is used for routing messages to the host, but also a natural identifier of the mobile node. The *home agent* (HA) is a router on the mobile node's home link. Any node communicating with a mobile node is called a *correspondent node* (CN). It may be either a stationary node or a mobile node.

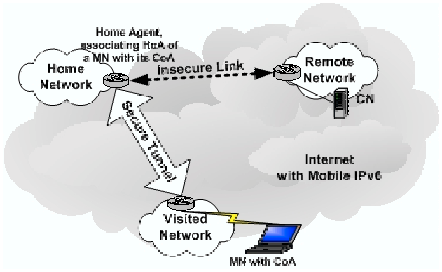


Figure 1: Mobile IPv6 architecture

An IPv6 address can be divided into two parts: the first part is the subnet prefix, and the second part is the interface identifier [8]. An IPv6 node periodically broadcasts Router Solicitation messages and waits for Router Advertisement messages. The IPv6 node can thus discover the subnet prefix from the Router Advertisement message, and then combine this subnet prefix with its own embedded Media Access Control (MAC) address to form a new IPv6 address. This feature in IPv6 is called *auto-configuration* [25]. An MN taking advantage of this technique is able to acquire a temporary local address, called the *care-of address* (CoA) without the use of a *foreign agent* (FA).

The HA remembers the association between the HoA and the care-of address of a mobile node, referred to as "*binding*". When a MN is away from its home network, all messages sent to its HoA will be routed to the MN by its HA. The HA keeps a binding list (called *Binding Cache*) so that it is able to know the current binding between a HoA and a CoA. An MN registers and updates its primary CoA associated with its HoA by sending a *Binding Update* (BU) message to the HA. Hence, wherever it is, any packet sent to the MN in its HoA will be forwarded to its CoA from its HA. Even if an MN moves to a new network, all packets of its existing transport-layer sessions will be routed to its new CoA. This is called bidirectional

tunneling in Mobile IP [11]. Moreover, Mobile IPv6 provides security between the HA and the MN by mandating a secure tunnel with IPSec. Packets from the CN are routed to the HA; then they will be encapsulated in the IPSec headers and destined to the MN. Correspondingly, the packets sent from the MN to a CN will be reversely tunnelled to the HA first. The HA decapsulates these packets, and then forwards them to the destination. The CN working this scenario does not need to implement the Mobile IP protocol. In addition, it is notable that the communication link between the CN and the HA is not secure.

In order to improve the efficiency of routing data packets, Mobile IPv6 defines a *route optimization* (RO) procedure, using which the MN is able to directly talk to its peers while roaming to different subnets, rather than indirectly communicating with its correspondents via its HA. Since the data packets will not travel via the home network by running the route optimization, this mechanism reduces the end-to-end packet delay. Route optimization also improves the reliability by reducing dependence on the home network, the HA and the path to and from it.

The route optimization process in Mobile IP results in security problems. Adversaries are able to send false binding update messages to CNs, in order to divert traffic to a third party who is intended to receive them, or to flood a third party who is unwilling to receive the packets. Adversaries also can hijack existing connections. Nikander *et. al.* [19] give an excellent overview of the intrusions that can be launched.

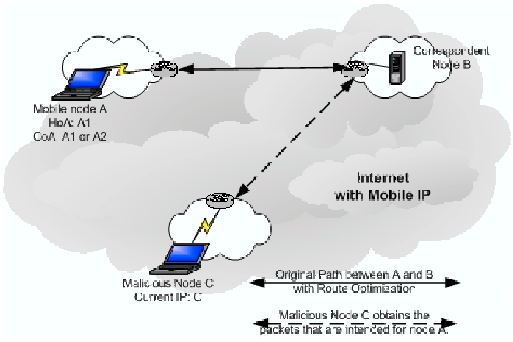


Figure 2: Basic address stealing

Figure 2 illustrates an address stealing scenario [19]. An MN A and a CN B are running route optimization. A malicious node C sends a false binding update message to node B, with the HoA $A1$ and the CoA C. If B believes the binding update, it will route all packets that are intended for A to C.

Figure 3 depicts a flooding intrusion by the traffic redirection [19]. While an MN A and a CN B directly transfer a heavy data stream, a malicious node C sends a false binding update message to B. The message tells B, "I am the MN A, and now my new CoA is address D." If the CN B believes the binding update, it will redirect the heavy data stream for node A to the victim that is located

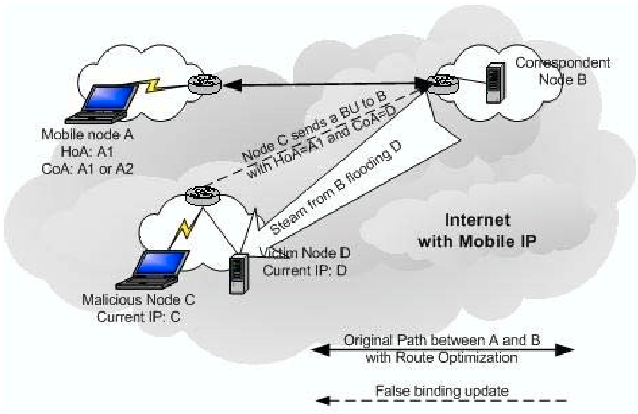


Figure 3: Flooding intrusion by the traffic redirection

at address D. If the malicious node C continues sending the acknowledgements to B, the heavy data stream will not stop flooding the node D.

As an alternative to this flooding intrusion, the adversary can request a heavy data stream from a CN B, and then send a false binding update message to B to change the current address to the target address D. As a result, if B believes the binding update, the victim D will receive unwanted packets from B. The adversary can replay the trick and redirect more unexpected packets from other CNs to flood the target D.

In this paper, we propose a hash chain based security mechanism to improve the security and performance of the return routability procedure in Mobile IPv6. Our fast authentication technique utilizes the hash chain element as an extra certificate to the mobile node in authenticating binding updates while running the routing process. In addition, by removing the necessity of the home test procedure in the RR test our mechanism can reduce the binding update latency. Given adversaries' different powers in the control of the communication path we analyze the security strength of our scheme. Furthermore, a performance comparison of our scheme and the original RR procedure is provided.

The rest of the paper is organized as follows. Section 2 provides the background and literature survey for Mobile IPv6 security. Section 3 describes the hash chain based security mechanism. Section 4 analyzes the security and performance of the new approach. Section 5 draws concluding remarks.

2 Background and Literature Review

In Mobile IPv6, there is a series of security designs aimed to protect the integrity and confidentiality of control and data messages, and RFC 3775 [11] designs the Return Routability (RR) Test, a lightweight and infrastructureless authentication mechanism, to resist most intrusions and alleviate the risk of network threats:

1) The IPsec ESP (Encapsulating Security Payload) secure tunnel between the MN and the HA protects the confidentiality and integrity of the information being transferred on this path. The HA is able to assure that all requests are from a legitimate MN, and vice versa.

2) Cryptographic functions are used to protect the integrity and authenticity of binding update messages. These functions depend on random numbers and keys exchanged between an MN and its correspondents.

3) The return routability procedure provides an infrastructureless method to ensure the binding update sent to the CN is a valid message from a legitimate MN. The binding management key K_{bm} is formed in this process.

4) The binding management messages guarantee the integrity of binding updates. These messages are verifiable by both parties, and they enable all operations to the bindings while an MN is running route optimization with its peers.

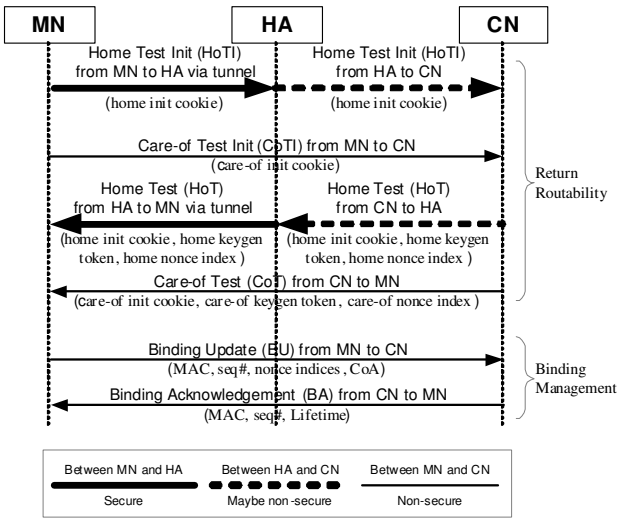


Figure 4: Return routability and binding management procedure

2.1 Return Routability Test and Binding Management

The return routability (RR) procedure is a mechanism used in Mobile IPv6 route optimization to assure an MN really owns the HoA and the CoA as same as what it tells its correspondents. Four messages are included in the procedure. *Home Test Init* (HoTI) and *Home Test* (HoT) messages must traverse the home network and be tunnelled between the MN and the HA. They are used to check the HoA, called return routability test for the HoA. *Care-of Test Init* (CoTI) and *Care-of Test* (CoT) messages are sent directly between the MN and the CN. They are used to check the CoA, called *return routability test* for the CoA. Figure 4 shows the return routability and binding management procedure [11].

In Figure 4, the return routability procedure can be regarded as two procedures, the home test procedure and the care-of test procedure. After the successful home test, the received home keygen token proves that the MN is the legitimate owner of the home address. After the care-of test procedure, the care-of keygen token shows that the MN is actually present at the new care-of address.

The return routability procedure is completed by the MN having received HoT and CoT. More importantly, at this time, the MN is able to compute the binding management key Kbm from the home keygen token and the care-of keygen token (Figure 4). During the binding management procedure, Message Authentication Codes (MACs) are computed using HMAC_SHA1 [11] by the binding management key K_{bm} to authorize the Binding Update (BU) message and the Binding Acknowledgement (BA) message.

For assuring security during the binding process, the return routability procedure is necessary to be run once before each binding management between the MN and the CN for any purpose, including updating, refreshing or deleting a Binding Cache entry (BCE). The return routability procedure, especially, the home test procedure in which messages have to travel through the HA, brings the long latency to the binding process. Therefore, if the home test procedure can be removed from the return routability procedure, the authentication mechanism will be more efficient.

Furthermore, the return routability does not protect against adversaries who are on the path between the HA and the CN [11]. As showed in Figure 4, the connection between the MN and the HA is secure and protected by the IPsec ESP [11]. The connection between the MN and the CN is non-secure because some links on the MN-CN path are wireless and vulnerable to intrusions. But an adversary cannot completely control the traffic between the MN and the CN because the integrity and authenticity of BU and BA messages are protected by their MACs. However, the connection between the CN and the HA is unprotected and may be non-secure. Although it is not easy for an adversary to insert malicious messages on the HA-CN path, the adversary that is able to only eavesdrop the home keygen token on the HA-CN path and can also easily steal the care-of keygen token by sending a fake CoTI message. From the eavesdropped home keygen token and stolen care-of keygen token, the adversary can compute the binding management key K_{bm} by which the intrusion can eventually completely control the traffic between the MN and the CN. This can lead to further intrusions. The eavesdropped home keygen token is valid until MAX_TOKEN_LIFETIME expires [11]. Thus the security vulnerability in the original return routability test is due to the non-secure HA-CN path.

2.2 Literature Review of Mobile IP Security

Many improvements are proposed to the Mobile IP security, but most of them are for improving security in Mobile IPv4. For example, a popular idea is to adopt public key cryptography and digital signature techniques [10, 17, 23]. Besides messages transferred among the mobile hosts, the network agents, and the correspondent nodes are encrypted using public or private keys. Some proposals suggest combining IPSec with Mobile IP [28]. The mobile node needs to establish IPSec tunnels in order to connect network agents and its correspondents, for the protection of control messages. Also some ideas recommend adding new Mobile IP featured devices and architectures into the Internet. For instance, the security gateway proposed in [9] enables bidirectional tunnels to other security gateways.

Until now, only a few proposals on improving the security mechanism of Mobile IPv6 other than the one proposed in the standard design from RFCs [2, 11] exist.

Vogt *et. al.*[26] propose an Early Binding Updates scheme to cut down the long latency associated with Mobile IPv6's home-address and care-off address tests. They only focus on improving the performance of the return routability test.

Hash functions and hash tokens have been used for real-time billing and payment in Mobile IP [24], for encrypting fields like MAC addresses in routing and security optimization in Mobile IP [6] and mobility management for Mobile IP [27].

Zhao *et. al.* [29] propose a hash-chain based mechanism to protect the return routability procedure. However, their security mechanism depends on the successful commitment of the first hash chain element. If the intrusion can steal the binding management key K_{bm} and can completely control the traffic between the MN and the CN, the intrusion can intercept and drop the first BU message which contains the first hash chain element and insert a malicious BU message instead with a fake hash chain element. In this way, the hash-chain based mechanism in [29] can be broken. In addition, since the hash chain element cannot be reused, the hash chain refreshment is required but has not been considered in [29]. Furthermore, their mechanism only reduces the signaling overhead when the MN does not change its location.

3 Improved Security Mechanism

In this paper, we propose a new hash-chain based scheme to improve the security and performance of the return routability test and binding management in Mobile IPv6.

3.1 Hash Chain Based Authentication Mechanism

Notations related to the hash chain are given as follows.

P_0 the root of the hash chain is a random number;

P_n the anchor of the hash chain;

H() a hash function;

P_i Any hash value of a hash chain, (natural number, $0 < i \leq n$), and $P_i = H(P_{i-1})$.

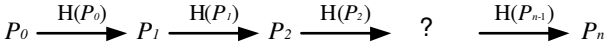


Figure 5: Forming a hash chain

Figure 5 shows how a hash chain is formed. A random value P_0 is generated at first, which is called the root of the hash chain. Then each element of a hash chain P_i is generated by applying the one-way hash function H() once to its pre-value P_{i-1}, i.e., $P_i = H(P_{i-1})$. A hash chain of length n can be formed by computing n hash elements sequentially. The last generated value, $P_n = H^n(P_0)$, is called the anchor of the hash chain.

An MN generates a hash chain $\{P_0, P_1, P_2, ...P_n\}$ before it builds a binding with a CN. In the authentication process, each hash element of the hash chain will be released sequentially. Due to the irreversibility of one-way hash function H(), any hash value $P_j (0 < j \leq n)$ that is eavesdropped by malicious nodes on the path to the destination will not help to generate any $P_i (i < j)$ in reverse. The CN remembers the last received hash value P_j. When the CN receives a new hash value $P_i (i < j)$ from the MN, it can apply the hash function H()$(j - i)$ times, and then compare whether$P_i = H^{(j-i)}(P_j)$ where $i < j$. The successful match shows that both P_i and P_j are from a same hash chain. It also authenticates the MN is the legitimate owner of the home address.

By using hash chain based authentication mechanism, the necessity of the home test in the return routability test can be removed so as to improve the performance of the return routability test.

One hash chain can only work for a home address-correspondent address pair, and cannot be used by any third party. Each hash value of the hash chain can be sent only once. When the hash chain is depleted after a long time use, the MN is required to refresh and generate a new hash chain.

Furthermore, hash functions are not time-consuming. They can be implemented either by using software or hardware. The computation by using hardware is easy and cheap, and offers extremely high performance. In [12], the throughput of SHA-1 by a hash processor is 114 Mbits/s. Sklavos and Koufopavlou [22] compare several implementations, and the throughput of SHA-1 is up to 233 Mbits/s.

3.2 Hash Chain Based Fast Authentication Scheme

Our fast authentication scheme makes use of the hash chain element in the binding update message as the second identifier other than the binding management key K_{bm}. In order to create a new binding, the MN must generate a hash chain and send the anchor value P_n to the CN. After a new binding is successfully established, each binding update message to the correspondent will carry an unused hash chain element that is from the same hash chain. Every time the correspondent receives a binding update, it will verify if the newly received hash value and the last received hash value are from a same hash chain.

In mobile IPv6, the tokens are normally expected to be usable for MAX_TOKEN_LIFETIME seconds. After MAX_TOKEN_LIFETIME expires, the MN should run the return routability procedure to get new tokens and computer new K_{bm}. Since the hash chain element is used as the second identifier, the home keygen token can be reused to remove the necessity of the home test in the return routability test. The specification limits the creation of bindings to at most MAX_TOKEN_LIFETIME seconds [11] can be removed.

Because all operations of manipulating the BCE are induced by binding update messages, the CN will take corresponding actions by processing each BU request. The CN operations on the BU and HoTI messages will be explained in detail in the next section. In the following section, we discuss procedures for creating, updating and refreshing a binding cache entry and the MN's operations in each procedure.

3.2.1 Creating a Binding Cache Entry or Refreshing the Hash Chain

Figure 6 depicts the procedure for creating a BCE, that is, that of building a new binding. For creating a binding cache entry, the last element in the old hash chain (P'_i, i) is of no use and set to (0, 0). Before the full return routability test, a new hash chain $\{P_0, P_1, P_2, ...P_n\}$ is generated at the MN. The anchor hash chain element (P_n, n) is send to the CN through the HA along with the HoTI. After a successful completion of the return routability test, the MN sends the BU to the CN to create a BCE. The MN receives the home keygen token and the care-of keygen token, remembers them, and then generates a binding management key K_{bm}. And the index of last used hash chain element, j is set to n, i.e., $j = n$.

In the first BU message, the hash chain element (P_i, i) is send to the CN. $i = j - 1$. And the index of last used hash chain element, j is updated as $j = i$.

When the index j is close to 1, it means the hash chain is going to be depleted after a long time use. The MN is required to generate a new hash chain for refreshing the hash chain. The refreshing procedure is almost same as the procedure for creating a binding cache entry. The only difference is that the CN is required to send the last element in the old hash chain (P'_i, i) together with the new anchor hash chain element (P_n, n). (P'_i, i) can work as the identifier for the new anchor hash chain element (P_n, n).

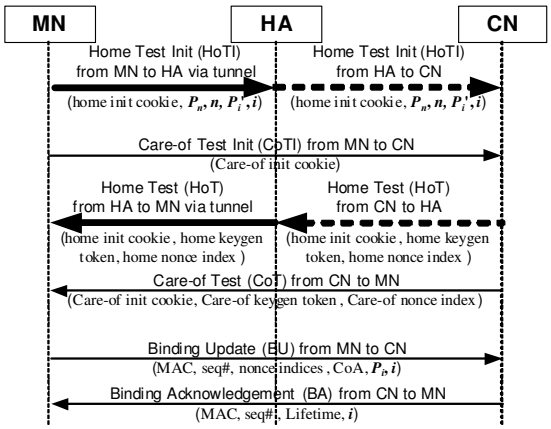


Figure 6: Creating the binding cache entry

3.2.2 Updating a Binding Cache Entry when MN Changes its Location

When the MN changes its location or its CoA is changes, a binding update is required to update a BCE. Different from the full return routability test necessary in Mobile IPv6 [11], only the care-of test is required. Figure 7 illustrates the procedure for updating a BCE.

After a successful completion of the care-of test, the MN sends the BU to the CN to update the BCE. The MN receives the new care-of keygen token and reuses the old home keygen token to generates a binding management key K_{bm}.

The MN sends a BU message to the CN, attaching an unused hash chain element (P_i, i) where $i = j - 1$ and j is the index of last used hash chain element. Also the index j is updated as $j = i$.

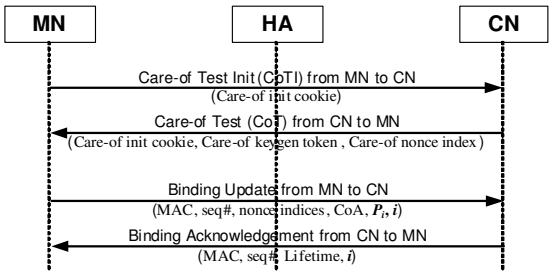


Figure 7: Updating the binding cache entry

3.2.3 Refreshing a Binding Cache Entry when MN Does not Change its Location

When the MN receives a Binding Refresh Request from the CN or its BCE lifetime is to expire, the MN should send a binding update message to its peer to renew the

lifetime of the BCE. The MN may refresh its BCE when it does not change its location or its CoA is not changed. Figure 8 shows the procedure for refreshing a BCE, that is, that of renewing a binding.

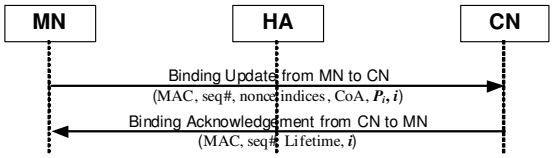


Figure 8: Refreshing the binding cache entry

In contrast to the full return routability test necessary before sending binding update to the CN in Mobile IPv6 [11], a hash chain element (P_i, i) in the BU message is used to identify the MN to the CN.

The MN sends a BU message to the CN, attaching an unused hash chain element (P_i, i) where $i = j - 1$ and j is the index of last used hash chain element. Also the index j is updated as $j = i$.

Once successful, the CN will update the lifetime of the BCE, and a binding acknowledgement should be sent back to the MN with a new granted *lifetime*.

3.3 CN Operations

Because all operations of manipulating the BCE are induced by binding update messages, the CN will take corresponding actions by reading each BU request.

3.3.1 CN Operations on the HoTI Message

When the CN receives the HoTI from the HA, the operation procedure is:

1) Check whether there is an existing BCE for this HoA.

2) If it does not exist, the CN is required to create a new null Binding Cache Entry and store the anchor hash chain element (P_n, n) in the BCE. Then the CN generates a home keygen token and send back the HoT message.

3) If it does exist and is a null Binding Cache Entry, the CN realizes that there is an adversary on the HA-CN path. However, the CN is not sure whether the currently received HoTI message or the previously received HoTI message is from the adversary. Thus the CN disables the null BCE and informs the HA the reason. Since the BCE has not successfully been created, the data packets are still forwarded to the HA and are not affected. The negotiated policy between the CN and the HA may decide to disable the route optimization for this HoA or not.

4) If it does exist but is not a null Binding Cache Entry, the CN is to refresh the hash chain as follows:

   - If (P'_i, i) comes from a same hash chain as the last received hash chain element (P_j, j), i.e., $i < j$ and $P_j = H^{(j-i)}(P'_i)$, the CN believes the new anchor hash chain element (P_n, n) is valid and update the last received hash chain element (P_j, j) to be (P_n, n).
   - If (P'_i, i) does not come from a same hash chain as the last received hash chain element (P_j, j), i.e., $i \geq j or P_j \neq H^{(j-i)}(P'_i)$, the CN realizes that there is an adversary on the HA-CN path and ignores the HoTI message.

5) If it does exist, the CN is also to updating the node key K_{cn} [11] and Nonces. Then the CN generates a home keygen token from new K_{cn} and new nonces and send back the HoT message. It can prevent the malicious node from re-using the old but unexpired keygen tokens.

3.3.2 CN Operations on the BU Message

When the CN receives the BU message from the MN, CN first generate K_{bm} and then verifies the MAC. If the MAC is valid, the CN performs an additional authentication check according to the received hash chain element (P_i, i) as follows:

1) Check whether there is an existing BCE for this HoA.

2) If it does not exist, the CN silently ignores the binding update.

3) If it does exist, the CN retrieves the last received hash chain element (P_j, j) stored in the BCE and verifies (P_i, i) received in the BU message with (P_j, j) as follows:

   - If (P_i, i) comes from a same hash chain as the last received hash chain element (P_j, j), i.e., $i < j$ and $P_j = H^{(j-i)}(P_i)$, CN believes this BU message is valid and update the last received hash chain element (P_j, j) to be (P_i, i).
   - If (P_i, i) does not come from a same hash chain as the last received hash chain element (P_j, j), i.e., $i \geq j or P_j \neq H^{(j-i)}(P_i)$, CN realizes that there is an adversary on the HA-CN path and can make a decision according to its local policy.

4) If the BU message is valid, check the binding *lifetime* in the BU message.

   - If *lifetime* $\neq 0$, the CN update the BCE with the binding *lifetime* and send back the BA message to the MN with a new granted *lifetime*.
   - If *lifetime* $= 0$, the CN delete the BCE and the specified care-of address is also be set equal to the home address [11]. The CN send back the BA message to the MN with a deleting notice.

4 Security and Performance Analysis

4.1 Security Analysis

The security strength of the original RR (return routability) test [11] and our hash chain based improved security mechanism is compared in this sub-section.

The identified threats on Mobile IPv6 Route Optimization are due to the spoofed binding updates. Existing intrusions described in [19] are as follows:

- Address stealing (Figure 2): An adversary could fabricate and send spoofed binding updates.
- Intrusions against secrecy and integrity: By spoofing binding updates, an adversary could redirect all packets between two IP nodes to itself.
- Basic denial-of-service intrusions: By sending spoofed binding updates, an adversary could redirect all packets sent between two IP nodes to a random or nonexistent address (or addresses) and stop or disrupt communication between the nodes.
- Basic flooding (Figure 3): By sending spoofed binding updates, an adversary could redirect a heavy data stream from the CN to the MN to the target address.
- Return-to-home flooding: By sending a spoofed binding deleting message to remove the binding from the Binding Cache, an adversary would redirect the data stream to the home network.

The security strength of the original RR test and our hash chain based improved security mechanism can be compared given adversaries' different powers in the control of the communication path. The control power of adversaries on the path can be categorized into three levels:

- *Level 1 control* - Basic sniffing: the adversary can only eavesdrop the traffic on the path and steal the secret keys.
- *Level 2 control* - Malicious messages spoofing: the adversary can spoof the legitimate address and insert malicious messages on the path
- *Level 3 control* or full control- Messages intercepting and modification: the adversary can intercept, drop and modify messages between legitimate users on the path.

Our security analysis also inherits all the underlying security assumptions of the original mobile IPv6 RR test [11], i.e., the MN-HA path is secure and protected by the IPsec ESP tunnel; the MN-CN path is non-secure and vulnerable by different level intrusions; the HA-CN path is unprotected and its security vulnerability can be used by adversaries (Figure 4). In most practical settings the network is likely to be more secure near the HA than near the MN [19]. The MN-CN path is more vulnerable than the HA-CN path, i.e., adversaries could have higher level control on the MN-CN path than that on the HA-CN path.

We present the security analyses and intrusions' assessment given adversaries' different powers on the HA-CN path and the MN-CN path as the following three scenarios.

4.1.1 Level 1 Control on the HA-CN Path and Level 2 Control on the MN-CN Path

The original RR test:
An adversary that is able to eavesdrop the HoT message on the HA-CN path to capture the home keygen token. On the MN-CN path, by sending a spoofed CoTI message, the adversary can also easily steal the care-of keygen token in the replied CoT message. From the eavesdropped home keygen token and stolen care-of keygen token, the adversary can compute the binding management key Kbm by which the intrusion can eventually send spoofed binding updates on the MN-CN path and launch all intrusions. Thus the security of the original RR test can be compromised due to the sniffing on the HA-CN path.

Our hash chain based mechanism:
Our hash chain based mechanism makes use of the hash chain element in the binding update message as the second identifier other than the binding management key K_{bm}. Even an adversary can capture the K_{bm} by the sniffing on the HA-CN path. However, the adversary cannot generate a valid hash chain element to spoof the valid binding update message. All intrusions on the MN-CN path can be detected and prevented.

4.1.2 Level 2 Control on the HA-CN path and Level 2 Control on the MN-CN Path

The original RR test:
The security of the original RR test can be compromised same as the first scenarios.

Our hash chain based mechanism:
An adversary has the power to send spoofed HoTI or HoT messages on the HA-CN path. If spoofed messages are sent after the BCE is created, the intrusion can be detected and prevented because refreshing the hash chain requires the last element in the old hash chain. If spoofed messages are sent during creating the BCE, the intrusion can be detected and the HA would be informed. The route optimization for the HoA may be disabled and all intrusions against Mobile IPv6 Route Optimization cannot be launched. However, it is possible but hard for an adversary to launch intrusions during creating the BCE because messages between the HA and CN is same as other normal Internet traffic and the adversary cannot

estimate when the MN will create the BCE.

4.1.3 Level 1/2/3 Control on the HA-CN Path and full Control on the MN-CN Path

The original RR test:
The security of the original RR test can be compromised same as the first scenarios.

Our hash chain based mechanism:
The security of our hash chain based mechanism can also be compromised. For example, in the foreign network, it is possible that the MN has connected to a rogue access point which is controlled by the adversary. The adversary can intercept and drop the valid BU message by controlling the rogue access point. By retrieving the valid hash chain element from this BU message, the intrusion can spoof a malicious BU message with the stolen K_{bm} key and send to the CN. The CN cannot detect this intrusion.

Thus, for the security analysis our hash chain based mechanism can improve the security strength of the Mobile IPv6 RR test when the HA-CN path is non-secure but the MN-CN path is not completely controlled by the adversary.

4.2 Performance Evaluation

This sub-section evaluates and compares the performance of the original RR test and our hash chain based mechanism.

For the simplicity of description, we adopt the following notations for the performance comparison:

- $\mathrm{T_{MN-HA}}$: the round-trip time on the MN-HA path protected by the IPsec ESP tunnel.
- $\mathrm{T_{HA-CN}}$: the round-trip time on the HA-CN path.
- $\mathrm{T_{MN-CN}}$: the round-trip time on the MN-CN path.

The original RR test:
For assuring security, the full RR test is necessary for every binding update. Thus the latency of each binding update procedure (Figure 4) can be approximated:

$$\mathrm{L_{ori}} = \max(\mathrm{T_{MN-HA}} + \mathrm{T_{HA-CN}}, \mathrm{T_{MN-CN}}) + \mathrm{T_{MN-CN}}.$$

Usually, the home test latency, $\mathrm{T_{MN-HA}} + \mathrm{T_{HA-CN}}$, is much larger than $\mathrm{T_{MN-CN}}$. The two-way message exchange between the MN and the HA is through the IPsec ESP tunnel. The IPsec processing may bring longer latency.

Our hash chain based mechanism:
The latency of the procedure for creating, updating and refreshing an BCE is different. The latency of the binding update during creating a BCE or refreshing the hash chain (Figure 6) can be approximated:

$$\mathrm{L^{creating}_{HashChain}} = \max(\mathrm{T_{MN-HA}} + \mathrm{T_{HA-CN}}, \mathrm{T_{MN-CN}}) + \mathrm{T_{MN-CN}}.$$

It is same as the latency $\mathrm{L_{ori}}$ since the full RR test is also required.

The latency of the binding update during updating a BCE (Figure 7) can be approximated:

$$\mathrm{L^{updating}_{HashChain}} = \mathrm{T_{MN-CN}} + \mathrm{T_{MN-CN}}.$$

It should be less than the latency L_{ori} since the home test is not required in the RR test. The latency of the binding update during refreshing a BCE (Figure 8) can be approximated:

$$\mathrm{L^{refreshing}_{HashChain}} = \mathrm{T_{MN-CN}}.$$

The binding update in our scheme when the MN does not change its location is at least 50% faster than the original binding update.

Thus, by removing the necessary of the home test procedure in the RR test our mechanism can reduce the binding update latency and improve the performance of the Mobile IPv6 RR test.

5 Conclusion

The vulnerability in the original RR test is due to the non-secure HA-CN path. If the HA-CN path is secure, all intrusions against Mobile IPv6 Route Optimization can be protected by the original RR test [11, 19]. Our hash chain based security mechanism can improve the security of the original RR test when the HA-CN path is non-secure. Our fast authentication technique utilizes the hash chain element as an extra certificate to the mobile node in authenticating binding updates while running the routing process.

In addition, by removing the necessary of the home test procedure in the RR test our mechanism can reduce the binding update latency.

Compared with the original RR test in Mobile IPv6, our improved security mechanism reduces the average latency of binding updates, and mitigates the threats especially from the adversaries on the path between the HAs and CNs. It reduces the burden of the HA, and decreases the probability of authentication packets traversing the home network.

Acknowledgements

We would like to thank the anonymous referees for their excellent comments and suggestions which have substantially improved this paper.

References

[1] B. Aboba and M. Beadles, *The Network Access Identifier*, IETF RFC 2486, Jan. 1999.

[2] J. Arkko, V. Devarapalli, and F. Dupont, *Using IPSec To Protect Mobile IPv6 Signaling Between Mobile Nodes And HAs*, RFC 3776, IETF Proposed Standard, June 2004.

[3] T. Aura, "Mobile IPv6 security," in *Proceedings of Security Protocols, 10th International Workshop*, LNCS 2845, pp. 215-228, Apr. 2002.

[4] H. Chen and L. Trajkovic, "Simulation of route optimization in mobile IP," in *Proceedings of 27th Annual IEEE Conference on Local Computer Networks (LCN'02)*, pp. 847-848, Nov. 2002.

[5] S. Deering and R. Hinden, *Internet Protocol, Version 6 (IPv6) Specification*, IETF RFC 2460, Draft Standard, Dec. 1998.

[6] A. Giovanardi and G. Mazzini, "Optimization routing and security features for transparent mobile IP," in *IEEE Global Telecommunications Conference (GLOBECOM'98)*, vol. 2, pp. 8-12, Nov. 1998.

[7] A. Harbitter and D. A. Menascé, "The performance of public key-enabled Kerberos authentication in mobile computing applications," in *Proceedings of the 8th ACM Conference on Computer and Communications Security*, 2001. (http://cs.gmu.edu/~menasce/papers/ccs8.pdf)

[8] R. Hinden and S. Deering, *Internet Protocol Version 6 (IPv6) Addressing Architecture*, RFC 3513, IETF Standards Track, Apr. 2003.

[9] A. Inoue, M. Ishiyama, A. Fukumoto, and T. Okamoto, "Secure mobile IP using IP security primitives," in *Proceedings of Sixth IEEE workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises*, pp. 235-241, June 1997.

[10] S. Jacobs and S. Belgaid, *Mobile IP Public Key Based Authentication*, July 2001. (http://www.watersprings.org/pub/id/draft-jacobs-mobileip-pki-auth-03.txt)

[11] D. Johnson, C. Perkins, and J. Arkko, *Mobility Support In IPv6*, RFC 3775, IETF Proposed Standard, June 2004.

[12] Y. K. Kang, D. W. Kim, T. W. Kwon, and J. R. Choi, "An efficient implementation of hash function processor for IPSEC," in *Proceedings of IEEE Asia-Pacific Conference (ASIC'02)*, pp. 93-96, Aug. 2002.

[13] S. Kent and R. Atkinson, *IP Encapsulating Security Payload (ESP)*, IETF RFC 2406, Nov. 1998.

[14] C. M. Kozierok, *The TCP/IP Guide*, No Starch Press, 2004.

[15] H. Krawczyk, M. Bellare, and R. Canetti, *HMAC: Keyed-Hashing For Message Authentication*, RFC 2104, Feb. 1997.

[16] D. A. Menascé, "Security performance", *IEEE Internet Computing*, vol. 7, no. 3, pp. 84-87, May-June 2003.

[17] M. Mufti and A. Khanum, "Design and implementation of a secure mobile IP protocol," in *International Conference on Networking and Communication (INCC'04),* pp. 53-57, June 2004.

[18] P. Nikander, J. Arkko, T. Aura, and G. Montenegro, "Mobile IP version 6 (MIPv6) route optimization security design," in *IEEE 58th Vehicular Technology Conference,* vol. 3, pp. 2004-2008, Oct. 2003.

[19] P. Nikander, T. Aura, J. Arkko, G. Montenegro, and E. Nordmark, *Mobile IP Version 6 Route Optimization Security Design Background*, IETF Internet Draft, draft-ietf-mip6-ro-sec-02.txt, work in progress, Oct. 15, 2004.

[20] B. Schneier, *Applied Cryptography: Protocols, Algorithms, AND Source Code in C*, Second Edition, Jan. 1996.

[21] *Secure Hash Standard*, FIPS PUB 180-1, National Institute of Standards and Technology, Apr. 1995. (http://www.itl.nist.gov/fipspubs/fip180-1.htm)

[22] N. Sklavos and O. Koufopavlou, "On the hardware implementations of the SHA-2 (256, 384, 512) hash functions," in *Proceedings of the 2003 International Symposium on Circuits and Systems (ISCAS'03),* vol. 5, pp. 153-156, May 2003.

[23] Sufatrio and K. Y. Lam, "Mobile IP registration protocol: a security attack and new secure minimal public-key based authentication," in *Proceedings of Fourth International Symposium on Parallel Architectures, Algorithms, and Networks (I-SPAN'99)*, pp. 364-369, June 1999.

[24] H. Tewari and D. O'Mahony, "Real-time payments for mobile IP," *IEEE Communications Magazine*, vol. 41, no. 2, pp. 126-136, Feb. 2003.

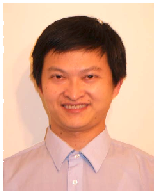
[25] S. Thomson and T. Narten, *IPv6 Stateless Address Autoconfiguration*, RFC 2462, IETF Standards Track, Dec. 1998.

[26] C. Vogt, R. Bless, M. Doll, and T. Kuefner, "Early binding updates for mobile IPv6," in *IEEE Wireless Communications and Networking Conference*, vol. 3, pp. 1440-1445, Mar. 2005.

[27] C.-C. Yang, J.-W. Li, and T.-Y. Chang, "A novel mobile IP registration scheme for hierarchical mobility management," in *Proceedings of International Conference on Parallel Processing Workshops*, pp. 367-374, Oct. 2003.

[28] J. K. Zao and M. Condell. *Use of IPSec in Mobile IP*, Nov. 1997. (http://www.csie.nctu.edu.tw/~jkzao/Publication/ietf-draft-mobileip-ipsec-tunnel-00.pdf)

[29] F. Zhao, S. F. Wu, and S. Jung, *Extensions On Return Routability Test In MIP6*, IETF Internet Draft, draft-zhao-mip6-rr-ext-01, Feb. 21, 2005.



Jing Li received the B.S. and M.E. degrees in computer science from Wuhan University, Wuhan, China, in 1996 and 2000, respectively. He is currently a Ph.D. candidate in Faculty of Computer Science at Dalhousie University, Halifax, Canada. His research interests include HSDPA networks, QoS in 3G networks, Mobile IP, WiFi and WiMAX security.

Po Zhang received the Master of Computer Science degree from Dalhousie University in Halifax, Nova Scotia, Canada in 2005. He has more than ten years of working experience in software development and system design. His major strength and interests are in the areas of security of Web applications.

Srinivas Sampalli is a Professor and 3M Teaching Fellow in the Faculty of Computer Science, Dalhousie University, Halifax, Nova Scotia, Canada. He has been actively researching in the area of security and quality of service in wireless and wireline networks. Specifically, he has been involved in research projects on protocol vulnerabilities, security best practices, risk mitigation and analysis, and the design of secure networks. He is currently the principal investigator for the Wireless Security project sponsored by Industry Canada. Dr. Sampalli has received many teaching awards including the 3M Teaching Fellowship, Canada's most prestigious teaching acknowledgement.