# Secure Route Optimization for Network Mobility Using Secure Address Proxying

Manhee Jo and James Kempf

DoCoMo Communications Laboratories USA, Inc.
3240 Hillview Ave., Palo Alto, CA 94304, USA
{mjo, kempf}@docomolabs-usa.com

## ABSTRACT

In this paper, we propose a secure route optimization mechanism for network mobility management. By means of the mechanism, the Mobile Router securely proxies the care-of address for the Mobile Network Node. The Mobile Router sends a Binding Update directly to the Correspondent Node on behalf of the Mobile Network Node. The Binding Update binds the Mobile Network Node's original address to the new care-of address. The Correspondent Node can verify that the Binding Update is sent from a node authorized to use the address. The Mobile Network Node authorizes the Mobile Router to proxy the address by using a Multi-key Cryptographically Generated Address to share the address ownership. In addition, through the binding procedure, the Mobile Router and the Correspondent Node securely exchange a session key, which enables a reduction in the handoff delay during binding update procedure.

***Keywords***: secure route optimization, secure address proxying, network mobility, multi-key cryptographically generated address.

## 1 INTRODUCTION

In order to support contiguous network connectivity, a mobile network should have mobility management functions. A mobile network can also contain multiple devices which may or may not have mobility functions themselves. In this situation, an entire network moving as a unit, such as a train, ship, aircraft and so on, dynamically changes its point of attachment to the Internet. Any node (host or router) located within a mobile network is called a Mobile Network Node (MNN) [1].

In this paper, we assume the MNN has no mobility management function itself. All mobility management must be handled by the Mobile Router (MR). The MR provides gateway functions to the MNNs, so that the MNNs send and receive packets through the MR. The MNN obtains an IP address within the mobile network and communicates with Correspondent Nodes (CNs) using that address. From a reachability point of view, an MR has to register its up-to-date location information, or a care-of address to a router in its home link named the Home Agent (HA). The care-of address is within the subnet managed by the wireless access network's Access Router (AR). The MNN's address appears to the Correspondent Node within the subnet managed by the MR's HA.

To provide mobility functions for Mobile Networks, the NEMO Basic Support protocol has been proposed [2]. It is an extension of Mobile IPv6 to allow session continuity for every node in the Mobile Network as the network moves to different points of attachment. The MR, which connects the Mobile Network to the Internet, does not distribute routes to the network infrastructure at its point of attachment. Instead, the MR runs the NEMO Basic Support protocol with its HA to establish a bidirectional tunnel between the MR and the HA of the MR. Thus, when the MR is away from the home link, the HA intercepts packets on the home link destined to the MR's home address, encapsulates them, and tunnels them to the MR's registered care-of address. The MR appears to the fixed network as if it is a host. Packets directed to and from the MNN appear in the HA's subnet and are tunneled by the HA to the MR, where they are de-tunneled and sent to the MNN.

In NEMO currently, however, routing is not optimized. The protocol maintains a bidirectional tunnel between an MR and the HA of the MR regardless of where the MR topologically is. All the packets between an MNN and the CN are propagated through the tunnel. This results in suboptimal routing (Figure 1).

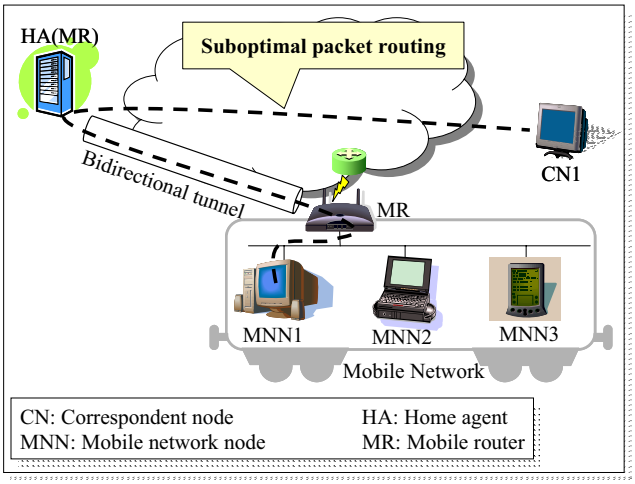


Figure 1: Suboptimal Routing Problem

Compared with the direct routing (i.e., over direct path between the MNN and the CN), suboptimal routing may cause the following problems [3]:

- Increased delay and additional total infrastructure load due to the longer routes,
- Increased packet overhead and processing delay due to packet encapsulations,
- Increased delay and reduced bandwidth efficiency due to the increased chances of packet fragmentation,
- Increased susceptibility to link failure between the MNN and the CN.

To solve this suboptimal routing problem, we try to make MR send up-to-date location information to the CN on behalf of the MNN. From security's viewpoint, if an unauthorized router sends bogus location information of the MNN to the CN, which could be a video streaming server, it will cause DoS attack or flooding attacks to victim nodes. Thus, the MR should be securely authorized by the MNN to send the MNN's location information. In this paper, we propose route optimization for NEMO based on secure authorization.

# 2 RELATED WORKS

## 2.1 Movement Notification-Based Methods

As a solution to provide an optimized route between the MNNs in the Mobile Network and the CNs outside the Mobile Network, Lee et al [4] propose a mechanism that the MNNs receive a prefix from an access router via the MR and send Binding Update (BU) messages to their own HAs directly by using the original Mobile IPv6 function. However, their proposal that every MNN sends a BU requires the MNNs to support mobility management software. In addition, their mechanism does not provide route optimization for MNNs without the Mobile IP function.

Cho et al [5] propose another route optimization mechanism for the Mobile Network called Recursive Binding Update. In their proposal, the binding information is sent from the MNN and the hierarchical MRs separately to the CN. The MNN's binding information is recursively collected and is finally merged into one Binding Cache Entry at the CN. Thus, after some convergence time, the CNs can maintain optimized routes to MNNs inside the Mobile Network. This achieves the optimized routing and, at the same time, reduces the end-to-end packet delay and therefore improves the throughput. However, in order for the CN to maintain the up-to-date binding information, the MNN must send BU messages to the CN every time the network moves. In addition, the MNN must be notified of its change in point of attachment by the MR every time the MR moves. From privacy standpoint, because the CN receives the binding information including all MRs' care-of addresses between the MNN and the CN, the internal topology of the mobile network is revealed to the CN, which is undesirable. In addition, there is no means to verify the MR's binding information.

## 2.2 Routing-Based Methods

Optimized Router Cache protocol (ORC) relies on specific routers called Correspondent Routers [6]. They are scattered over the Internet, hopefully near CNs or some traffic convergence point, and maintain a binding between the MR and the mobile network subnet prefix advertised by the MR in the mobile subnet. The routers intercept packets destined to HAs of the MRs. Those packets are encapsulated using their own routing information to be forwarded to the MRs directly. The routers advertise and update their routing information periodically by means of ordinary routing protocols such as IGP or EGP. Using this method, neither MNNs nor CNs need to be modified. This provides a transparent method by updating the routing information in the network infrastructure. However, since this method relies on the best-effort routing protocol, the deployment of the routers would be a critical problem and the performance of real time services might be an issue.

There is another method to support group mobility by using a routing protocol. Here, we call it BGP mobility [7]. In the BGP mobility, a Mobile Network, specifically an airplane, has its own network prefix. The devices in the airplane obtain their addresses using DHCP. When the airplane moves, the router in the airplane advertises its prefix to gateway routers on the ground by BGP announcement. Then, the gateway routers advertise the prefix information to other neighboring routers. By this method, since only the routers in the airplane have specific functions, network mobility can be achieved much more easily than with other complex mechanisms. However, as is the problem of BGP in general, the convergence time of routing information might be a problem. This kind of network mobility is a special case and it cannot be applied in general Mobile Networks which possibly need faster and more frequent handoffs. Also, devices in the Mobile Networks cannot receive traffic until they have initiated sending it, and the MR must be able to inject BGP routes.

# 3 APPROACH

For MNNs that do not support mobility management software, mobility must be hidden from the MNN. The methods in Section 2.1 make use of movement notification of the moving network and therefore require the MNN to have some parts of mobility management functions. Although decreasing the delay is one of the objectives of route optimization, the methods described in Section 2.2 do

not guarantee this because of the impact of these methods on the convergence time for routing information propagation.

The best way to achieve route optimization for NEMO is:

- The MR sends BU messages including the MNN's address and the MR's care-of address directly to the CN, and
- The CN appropriately generates a binding cache entry binding the MNN's address to the MR's care-of address.

However, in order for an MR to send a BU message instead of an MNN, the MR must be authorized by the MNN to change the routing for the MNN's address. That is, the MR must be authorized to act as an *address proxy* for the MNN. Moreover, the CN must be able to verify that the MR is authorized to act as a proxy for the MNN's address. Since secure address proxying is a more general operation than mobility management, it is more likely that an MNN would have the capability to participate in a secure address proxying protocol than that it would have mobility management functions.

Our approach is as follows. An MNN participates in a secure address proxy protocol on the MR's link that is an extension of the IPv6 Secure Neighbor Discovery Protocol (SEND) [9]. SEND allows nodes on the local link to establish secure bindings between the link address and IPv6 address, in other words, secure address ownership. The MNN shares address ownership of its address with an MR by means of a Multi-key Cryptographically Generated Address (MCGA) [8]. Generation of an MCGA requires two or more public keys rather than a single key Cryptographically Generated Address (CGA). Later, when the MR sends a BU message to the CN to update the MNN's address to the new MR's care-of address, it signs the message with its private key. The corresponding public key is also included in the BU message, so that the CN can verify the signature in the BU message. The MR also includes the MNN's public key, so the CN can verify the MCGA. The combination of the MCGA and signature establishes the authorization of the MR to update the binding. The next two sections describe the MCGA and the binding update procedure in more detail.

## 3.1 Multi-key Cryptographically Generated Address (MCGA)

In the SEND protocol [9], when a node configures its own IP address, the node uses a hash function over its public key and some auxiliary parameters to generate an interface identifier (i.e., the rightmost 64 bits in IPv6 address). This address is called a CGA (Cryptographically Generated Address) [10]. Signing all Neighbor Discovery messages with the private key, the node then can assert address ownership, because it can be guaranteed that only the node having the public key/private key pair can claim the address.

To generate an MCGA, we make use of public keys not only of the MNN but also of the MR. The detailed calculation procedure of the MCGA is described in Appendix. Figure 2 briefly shows the procedure to make and verify an MCGA ($A_{MNN1}$) by the MNN1 and the MR.

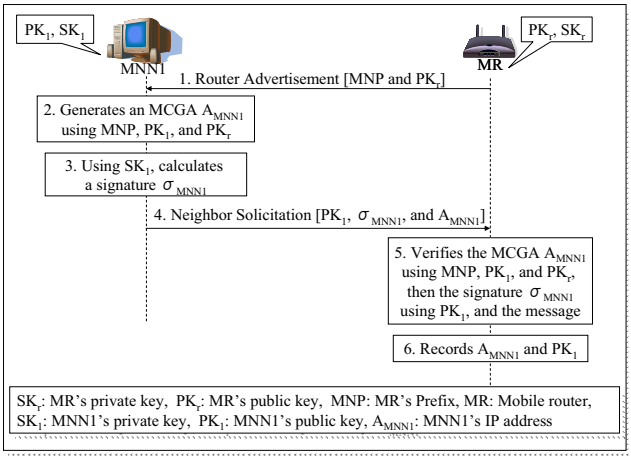


Figure 2: Multi-key Cryptographically Generated Address

In Step 1, MNN1 receives from the MR a Router Advertisement message including a Mobile Network Prefix (MNP) and a public key of the MR ($PK_r$) in CGA parameter option.[1] In Step 2, the MNN1 calculates the hash value of some auxiliary parameters in CGA parameters, such as modifier, subnet prefix, collision count, and the $PK_r$ as well as the MNN1's public key ($PK_1$). Then, the MNN1 utilizes the hash value as the interface identifier to the MNP to generate its own MCGA, $A_{MNN1}$. MNN1 determines the uniqueness of the address using the Neighbor Solicitation (NS) messages defined as part of the Duplicate Address Detection (DAD) procedure [11]. In Step 3, the MNN1 calculates a Signature ( $\sigma_{MNN1}$) over the NS message with its private key, $SK_1$. In Step 4, the MNN1 multicasts an NS message including the $PK_1$, the $\sigma_{MNN1}$, and the newly generated MCGA, $A_{MNN1}$. The MR receives the NS message in Step 5 and verifies $A_{MNN1}$ using MNP, $PK_1$, $PK_r$, and the other parameters for CGAs (modifier and collision count), then verifies the signature $\sigma_{MNN1}$ using $PK_1$ and the NS message. A failure at any step in the checks stops any further process. But if all the checks succeed, both the MR

[1] Note that the MCGA algorithm does not require any trust relationship between the MNN and MR. Since the MNN must know that the MR can be trusted to proxy its address, the MNN should only use the certified public key of the MR. The MR's certificate can be obtained using the SEND Certification Path Solicitation (CPS) and Certification Path Advertisement (CPA) message exchange [9].

and the MNN1 can now claim the address ownership of the same address. Therefore, the MR can send control messages concerning the IP address binding on behalf of the MNN1. Now, the MR records the $A_{MNN1}$ and $PK_1$ for future reference (Step 6).

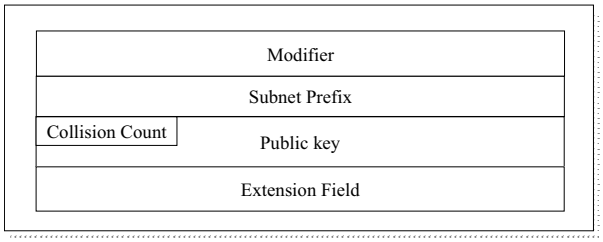


Figure 3: CGA Parameters

In addition to the two public keys, PK1 and PKr, the parameters in CGA option above include a modifier, a subnet prefix, a collision count, and optional extension field (Figure 3). The modifier is a predefined random bit value. As in [10], it is a 128 bit string. The subnet prefix is the mobile network prefix (MNP) advertised by the MR. The collision count is an eight-bit unsigned integer having 0, 1, or 2 used to recover from the address duplication. These parameters are included in a standard SEND CGA Parameters Option. Although the CGA Parameters Option only has an identified slot for one public key, the extra public key, $PK_r$, is put into the extension field. The standard SEND CGA calculation includes the additional parameters field, so there is no need to change the CGA verification procedure to accommodate MCGA. This is an important point for utilizing MCGAs in NEMO route optimization, because it means that CNs which use CGAs for Mobile IPv6 host to host route optimization can also utilize MCGAs for NEMO route optimization without any change in code.

## 3.2 Secure Route Optimization

Figure 4 shows the binding update procedure for secure route optimization. This is somewhat different from the original binding update procedure in [12]. This procedure is the same as the next generation route optimization protocol for Mobile IPv6 host to host route optimization described in [14]. Here, we call it initial binding update procedure. It includes a return routability procedure between the MR and the CN1, instead of between the MNN1 and the CN1. Return routability ensures that the node sending the BU is at the address claimed, and prevents redirection attacks.

At Step 1a, the MR sends the Home Test Init (HoTI) message to the CN1 through a tunnel to the Home Agent of the MR (HA(MR)). The source address is MNN1's MCGA. Since the prefix of the MNN1's address is maintained in the Prefix Table in the HA(MR), the HA(MR) can determine that the message is sent from a node in the Mobile Network. The destination address is the CN1's address. The HoTI message includes a home init cookie. At the same time, the MR sends the Care-of Test Init (CoTI) to the CN1 directly, not via the home agent (Step 1b). The source address is the MR's Care-of Address (CoA), and the destination is the CN1's address. The CoTI message includes a care-of init cookie.

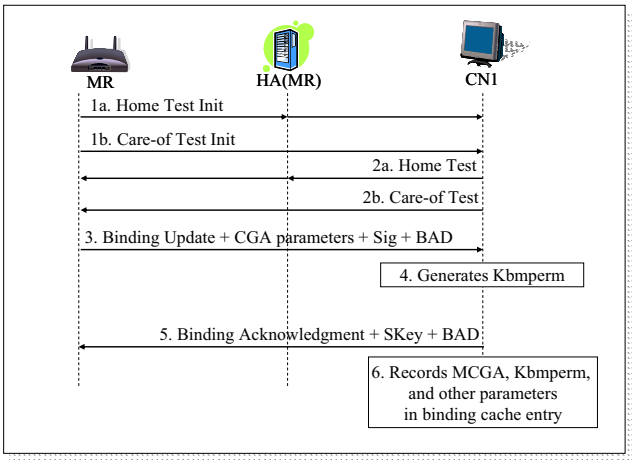


Figure 4: Initial Binding Update

At Steps 2a and 2b, the CN1 sends the Home Test (HoT) and the Care-of Test (CoT) in response to a HoTI and a CoTI, respectively. The source address of the HoT/CoT is the CN1's address. The destination address of the HoT and the CoT are the MNN1's MCGA and the MR's CoA, respectively. The HoT/CoT messages include home/care-of init cookies, a home/care-of keygen tokens, and home/care-of nonce indices. The home/care-of keygen tokens are calculated as in the RFC 3775 [12]. When a HoT message arrives at the MR, the MR checks if it is the first HoT message destined to the MNN1's MCGA. If so, the MR picks up only the necessary parameters and silently drops the HoT message. After the MR has received both the HoT and the CoT messages, the MR is ready to send a BU to the CN1. The MR, then, hashes the tokens together to generate a binding key Kbm, which is used for authorizing a BU message or a Binding Acknowledgement message (BA). Note that the MR must intercept the HoT and CoT messages directed to the MNN's MCGA.

At Step 3, the MR sends a BU message including MNN1's MCGA, a sequence number, home/care-of nonce indices, and Binding Authorization Data (BAD) along with additional options for CGA parameters and a signature. The CGA parameters include the MR's public key in public key field and the MNN1's public key in extension field (Figure 3). The MR calculates a signature over the BU message with the MR's private key. The signature is included in a RSA option, as in [14]. Using the Kbm described in the

RFC 3775, the MR calculates the authorization data to fill the BAD option.

At Step 4, the CN1 verifies BAD using Kbm, MNN1's MCGA and the signature using public keys, and then generates a semi-permanent security association key, Kbmperm, from the Kbm.

At Step 5, the CN1 sends a BA message along with a BAD and the Kbmperm. The BAD option is constructed using Kbmperm as described in the RFC 3775. The Kbmperm is encrypted with public key specified in public key field in the CGA parameters (i.e., MR's public key). The encrypted Kbmperm is placed into SKey option.

As a result (Step 6), the CN1 records the MCGA of the MNN1, the CoA of the MR, the public keys of the MNN1 and the MR, and Kbmperm. After that, the following state has been established in both the MR and the CN1:

- A standard Binding Cache Entry with a care-of address.
- A session key Kbmperm to be used for subsequent binding updates.
- The public keys and other parameters associated with the addresses.

Once the Kbmperm is securely exchanged between the CN1 and the MR, there is no need of full return routability procedure to negotiate Kbm again. Hence, the binding update procedure for subsequent movements can be simplified as described in [14]. Figure 5 shows still secure simplified subsequent binding update procedure.

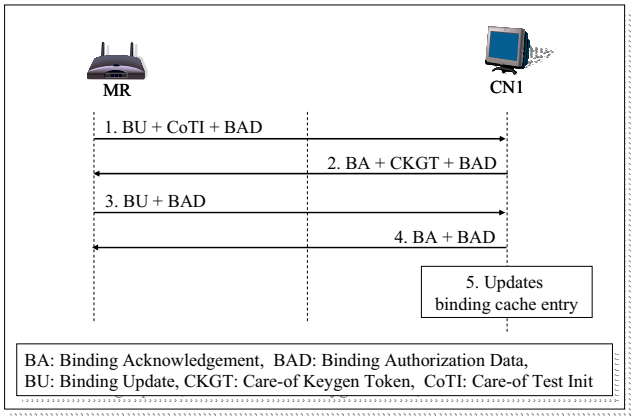


Figure 5: Subsequent Binding Update

When the point of attachment changes, the MR first configures its own CoA, which is not shown in the Figure. Then, to instruct the CN1 to redirect the packets to the new CoA, the MR sends a temporary BU to the CN1 including BAD and CoTI options at Step 1. The BAD option is calculated using the Kbmperm. At step 2, the CN1 sends back a temporary BA including the care-of keygen token in CKGT option.

At Step 3 and 4, the MR and the CN1 exchange a final BU and BA including BAD. The BAD is calculated using a new Kbmperm′ defined as HMAC_SHA1(care-of keygen token | Kbmperm). Since the Kbmperm is not known to other nodes, Kbmperm′ can be thought to be still secure.

At step 5, CN1 can successfully update the Binding Cache Entry. By means of this simplified binding update, it can greatly reduce the handoff delay. According to [14], the per-movement signaling is said to be reduced by 33%.

Now, the MR, on behalf of the MNN1, can send BU messages to the CN1 containing the MNN1's MCGA and the MR's new CoA every time the mobile network changes its point of attachment. After the Binding Cache Entry is updated successfully, a packet destined to the MNN1 includes the MR's CoA in the destination address and the MNN1's MCGA in the routing header. Receiving the packet from the CN1, the MR sets the destination address to the MNN1's MCGA and forward it to the MNN1. On the other hand, a packet destined to the CN1 from the MNN1 sets the source address to the MNN1's MCGA. Then the MR sets the source address to the MR's CoA and the home address option to the MNN1's MCGA. In order for a router to do this, we need to modify the MR's routing functions to change the address. In this case, the MNN's use of an MCGA provides the MR with authorization to proxy the address by changing it, unlike the usual use of address translation in IPv4 networks.

# 4 SECURITY CONSIDERATIONS

The purpose of the MCGA here is to allow the MR to securely proxy the address for the MNN. The claimed address ownership is verified by the signature. In this section, we consider several security issues: attacks from malicious nodes, and threat from malicious routers.

## 4.1 Attacks from Malicious Nodes

The MCGA is generated using mobile network prefix, the MNN's public key, and the MR's public key. The public keys are bound cryptographically to the address. Since those public keys are disclosed to anybody, an attacker can also collect the public keys to generate the address. Even though an attacker can collect those components of another MNN, it is impossible to collect any private keys corresponding to the public keys. The attacker cannot generate the signature and cannot impersonate the victim's address. Hence, even if an MNN receives the messages, such as Neighbor Solicitation or Neighbor Advertisement, from an attacker claiming the ownership of the same address, it can prove the authenticity of the messages by verifying the signature. Therefore, the attacker

cannot steal or spoof the existing address to redirect the victim's packets to somewhere else.

### 4.2 Attacks from Malicious Router

An attacker node (or router) can send bogus Router Advertisement messages including arbitrary or bogus prefix information and/or arbitrary or bogus parameters. This is an inherent problem in IPv6 not specific to NEMO. The problem can usually be solved by the trust relationship between the routers and the nodes. The trust relationship can be verified by checking the router's certificate. The trust relationship can be obtained by exchanging the Certification Path Solicitation (CPS) and Certification Path Advertisement (CPA) messages between the MR and the MNNs. If we assume that the MR and the MNN have the trust relationship with each other, so that the MNN can trust the MR, we can protect attacks from bogus routers, such as, bogus address configuration prefix advertisement, and parameter spoofing from the attacking nodes.

Even if a router has a certificate, it does not guarantee that the MNN's public key included in the Binding Update message is valid. If a certified MR turns bad, it could happen that the MR makes up a fake public key, generates an MCGA using the fake public key, and sends Binding Update messages signed using MR's private key to the CN. The CN, then, verifies the MR's signature attached on the Binding Update message, which turns out to be true, and the MCGA generated using MR's public key and the fake public key, which also turns out to be true. Thus, the CN cannot verify that the MNN's public key as well as the MNN itself is real. The validity of an MNN's public key can be made sure by using another certificate given to MR signed with MNN's private key. The certificate is sent to the CN along with the control messages (such as BU messages), so that the CN can verify the certificate using MNN's public key which is used for MCGA generation, which is not described in detail in this paper.

## 5 MCGA COMPUTATION OVERHEAD

Regarding the performance of our route optimization method, it could be argued that the MCGA's computation overhead is rather large. The MCGA calculation process is almost the same as that of CGA. The only difference for MCGA is to use one more hash operation of $PK_r$ and $PK_n$ to compute the address (Figure 2). Thus, the computation overhead of the MCGA over the CGA is expected to be trivial. In addition, the nodes or routers do not need to compute the MCGA very often. The MCGA is computed when a node is attached to an MR and when the MR sends initial BU messages to the CN. More time-consuming computation is due to the signature calculation and verification calculation. However, according to [15], the time for signature calculation and verification calculation are about 8.0ms and 0.2 ms, respectively. Moreover, this is, again, not the MCGA-specific overhead. All CGA based protocols, such as SEND, also have to experience the calculation overhead. Thus, the MCGA computation overhead over CGA can be ignored.

## 6 CONCLUSIONS AND FUTURE WORK

In this paper, we proposed a secure route optimization mechanism for the Mobile Network using Multi-key Cryptographically Generated Addresses (MCGA). By means of the MCGA, the Mobile Router can securely proxy the Mobile Network Nodes' address. The Mobile Network Node can securely authorize the Mobile Router to send Binding Update messages on behalf of the Mobile Network Node. In addition, both the Correspondent Node and the Mobile Router can securely share a session key for subsequent binding update. Receiving the Binding Update message from the Mobile Router, the Correspondent Node can verify the MCGA and the Signature, thereby verifies that the Binding Update message is sent from an actual address owner. Moreover, the signaling load for the return routability procedure is reduced by the simplified subsequent binding update.

From deployment's point of view, this mechanism can be easily implemented by extending IPv6 SEND. In this secure route optimization mechanism, however, the Mobile Router may be overloaded if it has to send out lots of Binding Updates. Ongoing work is attempting to determine the scalability limits of the mechanism. It might be solved by a hierarchical model deploying an anchor in the network.

In addition, we did not consider a Mobile Network Node with Mobile IPv6. If a Mobile Network Node has Mobile IPv6, it will try to send Binding Update to its own Home Agent. Since the tunnel between the Mobile Network Node with Mobile IPv6 and the Home Agent of the Mobile Network Node needs security association, the MR should be involved in this security association in order to successfully proxy the Mobile Network Node. This is another possible area of future work.

## REFERENCES

[1] J. Manner and M. Kojo, "Mobility Related Terminology," RFC 3753, June 2004

[2] V. Devarapalli, R. Wakikawa, A. Petrescu, and P. Thubert, "Network Mobility (NEMO) Basic Support Protocol," RFC 3963, January 2005

[3] C. Ng, P. Thubert, M. Watari, and F. Zhao, "Network Mobility Route Optimization Problem Statement," Internet Draft (work in progress)

[4] K. Lee, J. Park, and H. Kim, "Route Optimization for Mobile Nodes in Mobile Network based on Prefix Delegation," IEEE VTC 2003-Fall, October 2003, pp.2035--2038

[5] H. Cho, E. Paik, and Y. Choi, "R-BU: Recursive Binding Update for Route Optimization in Nested Mobile Networks," IEEE VTC 2003-Fall, October 2003, pp.2063--2067
[6] R. Wakikawa and M. Watari, "Optimized Route Cache Protocol (ORC)," Internet Draft (work in progress)
[7] B. Abarbanel, "Global Network Mobility," NANOG 31
[8] J. Kempf and C. Gentry, "Secure IPv6 Address Proxying using Multi-Key Cryptographically Generated Addresses (MCGAs)," Internet Draft (work in progress)
[9] J. Arkko, J. Kempf, B. Zill, and P. Nikander, "SEcure Neighbor Discovery (SEND)," RFC 3971, March 2005
[10] T. Aura, "Cryptographically Generated Addresses (CGA)," RFC 3972, March 2005
[11] S. Thomson and T. Narten, "IPv6 Stateless Address Autoconfiguration," RFC 2462, December 1998
[12] D. Johnson, C. Perkins, and J. Arkko, "Mobility Support in IPv6," RFC 3775, June 2004
[13] P. Nikander, J. Kempf, and E. Nordmark, "IPv6 Neighbor Discovery (ND) Trust Models and Threats," RFC 3756, May 2004
[14] J. Arkko, C. Vogt, and H. Haddad, "Applying Cryptographically Generated Addresses and Credit-Based Authorization to Mobile IPv6," IETF Draft (work in progress)
[15] J. Kempf, J. Wood, Z. Ramzan, and C. Gentry, "IP Address Authorization for Secure Address Proxying using Multi-key CGAs and Ring Signatures," IWSEC 06, October 2006

## APPENDIX

The actual address generation process at the MNN1 (Steps 2 and 3 of Figure 2) is as follows. The MNN1:

1 Calculates $PK_{temp}$, such that,
$PK_{temp}$=*SHA-1* ($PK_1$ | *PKr*),
where $PK_1$ and $PK_r$ are the public keys of MNN1 and MR, respectively.
2 Calculates
$H_2$ = *Leftmost 112 bits of SHA-1* (*modifier* | *subnet prefix=0* | *collision count = 0* | $PK_{temp}$)
3 Updates the modifier depending on the IPv6 security parameter value *Sec*
4 Calculates
$H_1$ = *Leftmost 64 bits of SHA-1* (*modifier* | *subnet prefix* | *collision count* | $PK_{temp}$),
where $H_1$ is the interface identifier to be concatenated to the *subnet prefix*
5 Executes DAD. On collision, updates *collision count* and repeats 4.

5.1 Before executing DAD, the MNN1 calculates the Signature over the NS message, say $\sigma_{MNN1}$. Then, the MNN1 sends the NS with $\sigma_{MNN1}$.

The verification process at any node which receives an MCGA (Step 5 of Figure 2) is as follows. The verifier:

1 Checks if the *collision count* is in the valid range, which is 0, 1, or 2.
2 Checks if the subnet prefix of the address is equal to that in the CGA parameter option field.
3 Calculates
$PK_{temp}$ = *SHA-1* ($PK_1$ | $PK_r$),
$H_{id}$ = *Leftmost 64 bits of SHA-1* (*modifier* | *subnet prefix* | *collision count* | $PK_{temp}$).
Then checks if $H_{id}$ is equal to interface identifier of the address.
4 Reads security parameter *Sec* from leftmost bits of interface identifier. Calculates
$H_{sec}$ = *Leftmost 112 bits of SHA-1* (*modifier* | *subnet prefix=0* | *collision count=0* | $PK_{temp}$).
Then depending on the *Sec* value, check if some predetermined part of $H_{sec}$ is equal to zero.
5 Verifies the Signature as described in [10].