A new Route Optimization protocol for Mobile IPv6 (ROM)

Christian Veigner[1] and Chunming Rong
Stavanger University College
Box 8002, 4068 Stavanger, Norway
christian.veigner@his.no, chunming.rong@his.no

Abstract-Mobile IPv6 is the mobility protocol of the IPv6 protocol. The mobility feature in IPv6 is supposed to be default, but still there are obstacles carrying out deployment. Great needs for this new Internet Protocol are security, scalability, efficiency and the lack of IPv4 addresses.

Deployment of this long needed new protocol has been delayed partly due to the route optimization problem of mobile nodes (MNs). Route optimization is the property of sending data via the shortest route, even when the MN is roaming, thereby making packet sending through a MN's home network for further redirection obsolete. Any other nodes communicating with a MN are referred to as corresponding nodes (CNs). Routing packets over the shortest route will both decrease the latency of data transmission and bandwidth consumption. In this paper, we suggest a new route optimization protocol for Mobile IPv6 (ROM), which we will show as a more efficient solution to this problem than other existing schemes.

Keywords-Mobile IPv6, route optimisation, authentication.

I. INTRODUCTION

Mobile IPv6 (MIPv6) [1] is standardized as a part of the Internet Protocol version 6 (IPv6) [2], and has a mandatory feature permitting MIPv6 nodes roaming from one subnet to another without disrupting their sessions during handover, where a handover is the change of connection from one network to another. It is crucial that handovers are authentic. IPv6 was supposed to be deployed in 2001, but partly due to authentication issues in MIPv6 route optimization, as shown in figure 1 and 2, the new IP protocol has not yet been deployed.
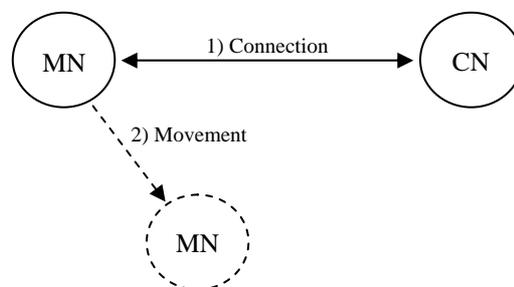

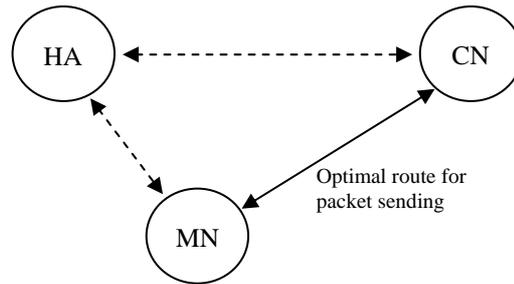Figure 1. Movement of mobile node (MN)

Figure 2. Route optimization

Consider a mobile node (MN) as shown in figure 1, having an ongoing session with a corresponding node (CN). If MN changes its point of location in the network, it is desirable to make the communication proceed between the MN and CN as shown in figure 2, without sending the packets through MN's home agent (HA) for further redirection to the correct MN. The HA is a node at MN's home link cooperating with MN in packet routing when MN is visiting a foreign network. In MIPv6, route optimization as shown in figure 2 is the default mode; every IPv6 node has to support it.

There are actually no distinctions between a MN and a fixed node in IPv6, which enhance the necessity of completing the work of MIPv6 even more.

How to make the flow of data from a MN to a CN and opposite without sending the packets through the HA has been a topic of recent research. There are several proposals for this type of management, e.g. RR [1] CGA [3], CAM [4], HIP [5], ABK [6], BAKE/2 [7] and IPsec [8], where each of these proposals suffers from different weaknesses. CGA is patented, CAM is similar in many ways to CGA, but has security $O(2^{62})$, assuming a brute force attack. This level of security may be enough considering most attackers currently available and affordable technology, but the design of IPv6 should be secure for more than the next decade. HIP, ABK, BAKE/2 and IPsec assumes the existence of additional infrastructure, and will not very easily be integrated in existing networks for managing Mobile IPv6 handovers in a secure and scalable way. The protocol chosen by IETF is the Return Routability (RR). It has some known weaknesses, though it does not rely on any additional infrastructure.

Considering the RRv3 protocol, it might be too time consuming when a MN suddenly has to initiate a handover to a new subnet, which may result in the user suffering from lack of seamless handover.

ROM is a protocol initiated by a mobile node (MN). Its goal is to prepare the corresponding nodes (CNs), which the MN is currently communicating with, of MN's change of location prior to the actual movement. When MN has changed its point of location, only one single message to each of the MN's CNs is required to make verifiable new bindings. This will increase the probability of achieving seamless handover procedures.

In Mobile IPv6 the MNs must always be reachable by their IPv6 home addresses. CNs are initially unaware of the current location of a MN, and will use the MN's home address (HoA) when transmitting a packet. In this way, all packets will be routed

through the HA. This HA reroutes the packets intended for a MN to the MN's new location. To manage this, the MN has to register its foreign address at the HA before the HA is able of rerouting incoming packets.

In time users will demand uninterrupted sessions when moving securely from one subnet to another. This is the motivation for implementing ROM.

Constructing a location management protocol we believe in the importance of scalability, entailing the importance not making the protocol rely on additional infrastructure as certificate nodes, trusted third parties or any other infrastructure that will be both cost consuming and uneasily scaled. Security issues of the protocol are also very important.

II. RETURN ROUTABILITY (RR)

This is the protocol currently considered by IETF for securely managing mobile nodes (MNs) binding updates (BUs) sent to corresponding nodes (CNs) and home agents (HAs). RR assumes the existence of an IPsec Security Association (SA) between a MN and HA. The communication between MN and its HA is secured by IPsec ESP.

A. RRv3

The most recent version of RR is version 3 (RRv3) [1]. In this scheme, when updating a binding with a CN, the MN has to send three and receive two messages from CN in three different stages, as shown in figure 3. This is all done in turn after MN has sent a binding update (BU) message to its HA and received binding update acknowledgment in return (BUAck). In RRv3, all the explained messages must be sent when the MN has arrived at its new location. This is a great threat for the seamlessness of handover procedures. The reason for sending messages via different routes is the belief in the hardness of an attacker monitoring two different dynamically changing routes at the same time.
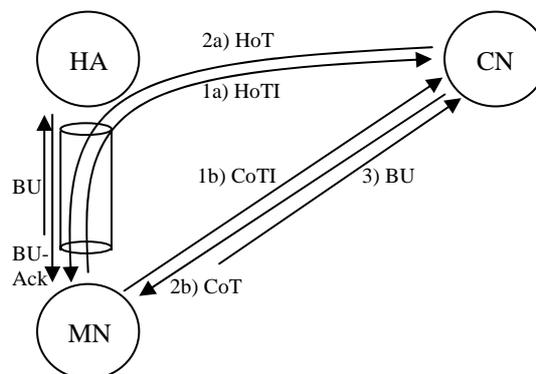

Figure 3. RR version3

If a MN moves quickly from one subnet to another, and suddenly looses its connection with the old subnet, it will be out of connection until the RRv3 protocol finishes executing at the new network. Constructing a protocol reducing the binding

latency reduces the possibilities of noticeable handover procedures. We believe, in time, users are going to demand non-interrupted communication with their CNs even when they are in transit, which motivates the construction of a quicker handover protocol as ROM. Another mayor problem in RRv3 occurs if the HA is off-line at the moment MN tries to initiate a handover. If the HA is down, there is no way MN can continue its session with a CN from the new location without subscribing to a new HA before initiating the RRv3 protocol. This problem is eluded in the ROM protocol design.

III. ROM

A. Overview

The overall goal of the ROM protocol is to minimize the elapsed time from a MN in subnet transit realizes it has to connect to its current CNs via a new subnet, to the new connection is ready for use. If the MN is moving quickly from one subnet to another and/or the signalling strength from the old subnet is rapidly decreasing, there may be no time sending messages back and forth establishing a new verifiable connection with the CNs from the new subnet, and still provide seamless handover to the user. As mentioned in the introduction, using the ROM protocol, a MN only has to send one message to each of the CNs which it has ongoing sessions to re-establish a verifiable binding when changing its point of location. This is a huge improvement from RRv3 where MN first has to bind to its HA, which may be off-line, then exchange four messages with each CN of current interest, and then finally send a BU message to each of these CNs.

In section IV we analyse different attacks possible when offering mobility, and show to which extent our protocol, compared to the RRv3 protocol, withstand these attacks.

Briefly, the ROM protocol is to be used by a MN for sending a hash value via the HA to each of its currently used CNs. This is done before movement. Each of these hash values should be unique. When changing location, MN must send a binding update (BU) message directly to each of these CNs if it wants to continue its sessions. A CN is now capable of verifying the new binding due to MN's knowledge of the nonce value used to generate the hash. This nonce value must be included in the BU message. We will now explain the ROM protocol in more detail.

In ROM, as in RRv3, we assume there exist an IPsec SA between MN and HA.

A MN may want to establish a binding with one or several previously unused CNs. If the MN is away from home, our solution is to make MN send a list to its HA containing IPv6 addresses of these CNs and corresponding hashed nonce values, see message 1 in figure 4. These hash values are generated by MN hashing different nonce values. HA sends the correct hash value to the correct CN, message 2a in figure 4, one hash value to each CN, explained in further details later. MN will then, when binding, send a BU message directly to the CNs it wants to bind with. A BU message will thereby be verified at a CN by the nonce value included in the message. A one-way hash function is needed.

The 2a–2c sequence of messages is only intended to prepare the CNs of MN's next BU message. A BU message may be sent by MN from the current location establishing or renewing its binding, or it may be sent from MN's new location.
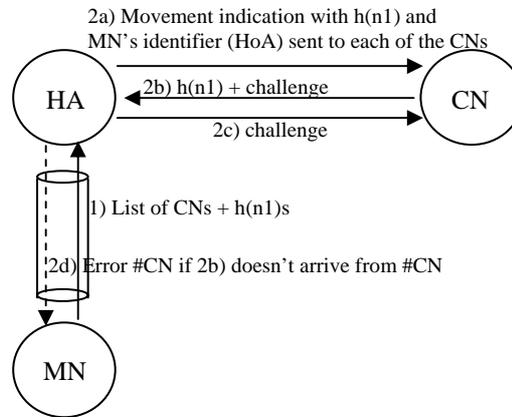

Figure 4. Preparation of CNs

When a CN gets a BU message from the MN (depicted later) containing a nonce value, the corresponding hash value used for verification will be obsolete and never used again. The CN must now be sent a new hash value as quickly as possible via MN's HA to be able of verifying a future BU message from MN.

New hash values must also be sent at given intervals to currently used CNs, even though they are in possession of hash values not yet used for verification and currently usable for verifying MN's next BU message. Sending of new hash values to such CNs could occur at given intervals, depending on the security of the hash function. If it is truly one-way the valid time may be extended. The reason for generating new nonce values and sending their corresponding hash values, even to a CN still in possession of a valid hash value usable for verification of the next BU message, is a security matter. The hash value can be detected by a passive attacker at the HA–CN link. This attacker can, if the nonce or a valid value is found, turn active and create a false BU, redirecting MN's incoming traffic.

In the general case, when the ROM protocol is used for speeding up a handover, message 2a in figure 4 is just an indication to a CN of MN's next movement. This message must also include MN's identifier (HoA), the home address of the MN, in addition to the hash value. The CN must store MN's home address, the hash value, and MN's verified care-of address (CoA) in a location table. The CoA is the IPv6 address a MN is associated with at its foreign subnet. The HoA and hash are stored at CN when receiving message 2a and 2c, figure 4, but the CoA is stored at CN when CN later receives a verified BU message from MN. CN does not change its location table for MN's CoA until a verified BU from the MN is received, thereby making it impossible for an attacker hijacking another MN's session without knowledge of the secret value used for generating the hash. Also; the hash value (h(n1)) at CN will not be changed until the 2a–2c procedure has been completed. This assures CN that the new h(n1) is sent from MN via its HA. It is the MN's home network that is authenticated by this message exchange.

The 2b message in figure 4 must contain the h(n1) value, making the HA able determining if it has sent the 2a message to this node beforehand. 2b should also include a challenge that must be returned by the HA. This authenticates the MN's home network. However there might be a man-in-the-middle attack, but this is as in the RRv3 protocol not considered a problem of the mobility protocol. The aim of this protocol is not solving the problems of the entire IPv6 protocol, but only to avoid new ones due to the mobility feature.

If HA doesn't receive 2b in return of 2a within a predefined short time, HA should send error message 2d as shown in figure 4 to the MN, notifying MN which CNs are unavailable/off-line, and not necessary sending BU messages to until online again.

The BU message (figure 5) from MN to CN will verify the previously received h(n1) at CN, by containing the correct nonce value. Inclusion of the HoA in the BU message tells the CN for which MN it should attempt verify the new binding.

The hashed nonce values sent to a CN via MN's HA should be unique. Each hash should also be different from the other hash values sent to the other CNs that MN has in its list of recently used CNs. This is a security property. An attacker may log all the different hash values sent via MN's HA, and over a long period of time find valid nonce values, or maybe getting the nonce value sent from MN to a CN when MN is verifying its hash value. It is assumed hard for Eve monitoring both these links, but not necessarily if Eve is close to CN, then the two links HA–CN and MN–CN may coincide. RRv3 bases its security on the hardness of this property. If MN uses the same hash value again, Eve can check its list, possibly finding a valid nonce for the hash value, and then hijack MN's session before MN has sent its binding update message to the CN.
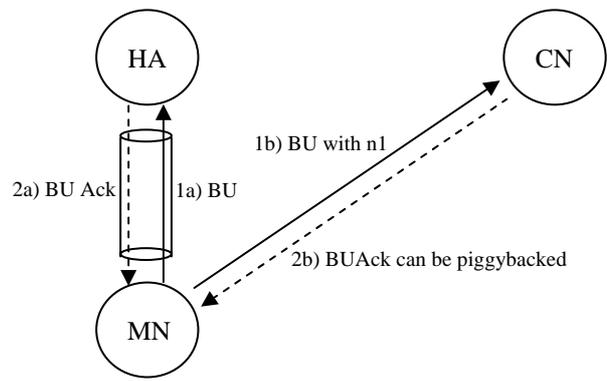

Figure 5. BUs from MN

B. ROM improvements compared to RRv3

The messages in figure 4 may be sent before MN moves to its new location. If the MN moves to a new location and sends BU messages to its CNs, then the MN must once again send a list to HA providing the CNs with new hash values usable for verifying MN if it changes its point of location once again, or if the binding valid time is to expire and the need for sending a new verifiable BU message from MN's current location arises.

When MN has lost a connection or changes its point of location, it has to configure a new IPv6 address at the new subnet, as in RRv3, but is spared from sending the two messages via different routes (figure 3) and receiving the two messages also via different routes before it can send BU to its CN. In RRv3, all this has to be done after the MN successfully has updated its HA. The reason in RRv3 for sending the four messages (HoTI, CoTI, HoT and CoT) is to get insurance that MN is where it claims to be. RRv3 will in this way be secure against attacks from elsewhere the HA-CN link, but it is still a possibility for an attacker at the HA-CN link to launch a redirecting attack [9]. This is though a huge improvement, reducing redirecting attacks from the entire Internet to the HA–CN link.

The ROM protocol also uses two different routes, but this is not done as a security cause as in RRv3, but for the sake of faster handovers. One route is used for verification of the belonging home link and for sending of hash values. The other route is used for managing the actual handover when necessary as quickly as possible.

As in RRv3, redirecting attacks where Eve is not at the HA-CN link is also avoided in our scheme. ROM in fact narrows the attackers range to launch its attack from the HA-CN link as in RRv3, to the necessity of being within the home link of the attacked MN. In message 2b, figure 4, CN checks if the hash is real and is sent from the given home link by challenging the HA.

If the HA suddenly is not able to act as MN's HA, e.g. due to down time, MN's session with a CN when changing location is not interrupted, assuming CN has a valid hash value. MN's negotiation with the home network getting hold of a node willing to act as its HA does not lead to any further delays in the connection re-establishment with the CN. This is a great advantage compared to RRv3. Seamless handovers are not dependent of the immediate cooperation of the HA.

IV. ATTACKS ON ROM

Constructing a secure protocol it is important considering different possible attacks. As in RRv3 we will in ROM not consider attacks also possible in fixed IPv6, e.g. man-in-the-middle attacks. We only attempt avoiding attacks possible due to the mobility feature.

A. Redirecting MN's sessions

In a redirecting attack Eve (the attacker) must generate her own nonce and corresponding hash value as a part of the attempt redirecting MN's incoming traffic. This is the only feasible solution as long as the MN uses a one-way hash function.

Eve sending a message to CN impersonating another MN's HA is not possible having the 2b and 2c messages in figure 4 verifying the location of HA and the fact that HA actually sent the hash originating from its corresponding MN. The message 2b contains the received hash value and a challenge, and in return HA will answer the challenge and indicate to CN whether CN's received hash is legitimate. Eve will not be able of impersonating other HAs and sending false hash values CNs for which Eve has the corresponding nonce values, unless Eve is a node at the MN's home network.

These nonce values could then be used for sending a false location message (BU) as if coming from MN, resulting in a redirecting attack on MN.

B. Flooding attacks

The CNs in this protocol must have a table as in figure 6. If e.g. a PDA contacts a MN, and the MN is currently not at home, the PDA must store the location information it receives in its table. This may sound as a drawback, but managing route optimization a node has to have a location table. The restrictions and properties of this table will be subject to further research.

HoA	h(n1)	CoA
⋮		

Figure 6. Location table at CN

This table consists of MNs HoA addresses, hashed values and MNs CoA addresses that this CN currently has contact with. There is a possibility of flooding attacks against this table at CN, possibly resulting in DoS. We may think of an Eve sending movement indication message (2a, figure 4), resulting in CN storing the HoA and h(n1) values. This only occurs if Eve is responding to the 2b with the 2c message.

1)   If Eve is impersonating another HA and sends a CN new information (HoA and h(n1)), CN will not get the expected 2c message from this HA (possibly error indication), and will in turn reject the information sent by Eve.

2)   A flooding attack against CN's buffer may be launched by an Eve who uses her own address as a HA and sends a 2c message in reply of the challenge from CN as mentioned. This Eve is now able of sending many different HoAs and h(n1)s to CN. CN will store these values in its table. This can be considered a problem, but may be minimised by restricting the number of MNs a HA can have in its domain. Another possibility is to restrict the number of binding nodes allowed by CN from a given network. A distributed attack is still possible, even though these rows will be deleted after a short time when not used. Anyway, this seems to be the most critical part of our protocol. It may lead to DoS attacks, and should most definitively be topic of further research. It must be added; the existence of a table at CNs for managing route optimization can't easily be omitted.

3)   Eve may also flood CN's buffer by sending lots of BU messages from different false addresses. CN has to remember these BUs in case it receives an authentic location change indication message via a HA, message 2a in figure 4, shortly after from a MN establishing route optimization, where the MN tries to bind before the h(n1) value has reached the CN. This may occur due to extra long latency in the MN-HA-CN route. Preventing an attack, CN must drop the BUs that it can't yet authenticate after e.g. 3 seconds. In this case, if a non-fraudulent node's BU message was dropped, because of delays in the MN–HA–CN route, it has to retransmit its BU message. MN is almost certain of CN's availability when not receiving an error message from HA indicating that CN is unavailable/off-line, message 2d in figure 4.

C. Bombing attacks

We may consider a scenario where an attacker is not impersonating anyone, but connects to a CN offering video streaming services. The attack is to redirect this stream of data to a victim node anywhere in the entire Internet. This attack is functional because the attacker may generate a nonce value and a corresponding hash value. Eve will initiate the sequence getting CN ready for her spurious movement. The CN is now provided with Eve's correct HoA and a h(n1) value waiting to be verified by a BU message sent from Eve from her new location. Forging the BU message using the HoA of Eve and inserting the verifiable nonce value, Eve uses the victim's address as the CoA. CN will now redirect its streaming service to the specified address in the source address part of the BU message's IPv6 header, the victim MN.

This bombing is only efficient if the attacker at some point has redirected the stream from itself. By this strategy gaining information about the sequence numbers in the incoming TCP packets. The attacker has to send false TCP acknowledgements back to the streaming provider once in a while, inserting the address of the victim as the source address in the packets, to make the attack continual. The attacker only has to send one acknowledgement message per TCP window to realize its intention.

Mobile IPv6 defines a new routing variant, type 2 routing header, used for sending packets optimized to a MN from a CN. The bombing, as described, seams efficient, and can't easily be stopped by the attacked node. In this scenario, the attacker must use its own HoA address as a home address option and the CoA address of the attacked node as the source address part of the BU message to fulfil its bombing intension. The CN will now redirect the stream to its victim. When receiving the stream, the attacked node will drop any packet at IP level when detecting that wrong HoA is used as the final destination. This is due to the routing header type 2 used for route optimization, see [1] for explanation of the use of routing headers. If all the packets of the stream are silently discarded, it is not possible for MN processing the attack at a higher level generating a message telling CN to halt the stream.

If possible somehow, informing the CN to stop the bombing, the solution depicted in figure 7 might be a solution.
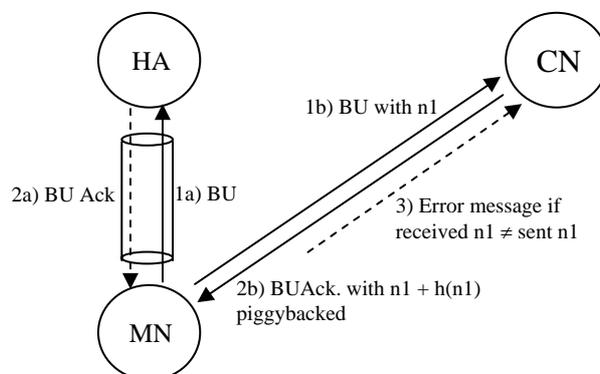

Figure 7. Authentic BU message

The CN could piggyback the received nonce value to the MN, included in the mobility options of the returned BUAck message (figure 8). By IETF the reason for having the mobility option as a part of the BUAck message is to allow future extensions to be defined. If the MN didn't sent the nonce an error message to CN should be sent. The added time penalty is limited using piggybacking.
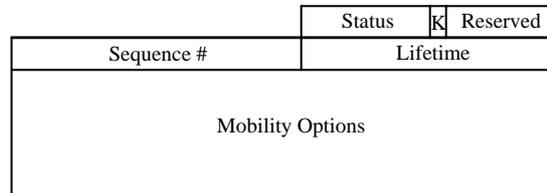
	Status	K	Reserved
Sequence #	Lifetime		
Mobility Options			

Figure 8. BUAck message format

MN may also overcome an attack where Eve located at MN's home link updates MN's CN with a false h(n1) value. This attack can be handled if CN sends the MN the currently stored h(n1) as a part of the mobility options in the BUAck message. This should only be done if the stored h(n1) at CN is different from the hash value generated at CN from the received nonce value when MN is binding. Sending the stored h(n1) value back to MN, MN knows if it has been attacked by a node at its home link having changed the h(n1) value, which is possible in our scheme, but only from within the home link. The MN must now send a new hash value via HA to CN, making another attempt binding with CN.

D. Amplification attacks

In our protocol no attacker is able of sending one message and thereby making the receiver send more than one message. HA and MN has a security association, and neither HA nor MN can be used for launching this attack against one of the other nodes while sending one message to them only results in one message sent from them. Also; considering CN (figure 4 and 7) we can see that sending one message to it only results in one message sent from it. The attacker can thereby not amplify its attack.

V. CONCLUDING REMARKS

Our ROM scheme is intended to make the handover in Mobile IPv6 more seamless than RRv3 manage, i.e. to speed up the actual handover and also provide similar security characteristics. This is attempted done by using some info sent to the CNs before changing location. Then, when changing location, only send one verifiable BU message to each of the CNs.

The main drawback of the ROM protocol as we see it is the allocation of a table at CNs, which may be any node. A CN will store states in certain columns of the table after executing the three-way handshake with a HA. This is a potential for launching DoS attacks filling it.

Redirecting attacks as mentioned in [9] at the RRv3 protocol from the HA-CN link is in ROM limited to the home link of the attacked node.

The home agent in our scheme is also a less crucial node, and handovers can be carried out seamlessly even if the HA is down. This is a great improvement compared to the RRv3 protocol.

As we all know, bandwidth in mobile and wireless networks is very unpredictable and often low. This is the main benefit of the ROM protocol; the reduction of the amount of messaging necessary to re-establish the connection when arriving at a new network.

It is also very important that we understand all the threats new technology creates before a possible deployment.

VI. REFERENCES

[1] D. Johnson, C. Percins, J. Arkko, "Mobility Support in IPv6", Jun. 2004.

[2] S. Deering, R. Hinden, "Internet Protocol, Version 6 (IPv6) Spec.", Dec. 1998.

[3] T. Aura, "Cryptographically Generated Addresses (CGA)", Feb. 2004.

[4] G. O'Shea, M. Roe, "Child-proof Authentication for MIPv6 (CAM)", 2001.

[5] P. Nikander, J. Ylitalo, J. Wall,
  "Integrating Security, Mobility and Multi-homing in a HIP Way", 2003.

[6] S. Okazaki, A. Densai, C. Gentry, J. Kempf, A. Silverberg, Y. L. Yin, "Securing MIPv6 Binding Updates Using Address Based Keys (ABKs)", Oct. 2002.

[7] M. Roe, T. Aura, G. 'Shea, J. Arkko,
  "Authentication of Mobile IPv6 Binding Updates and Acknowledgments", 2002.

[8] IETF, "IP security Protocol (IPsec)".

[9] R. H. Deng, J. Zhou, F. Bao, "Defending Against Redirect Attacks in Mobile IP".