Network Working Group
INTERNET-DRAFT
Expires: September 7, 2005
Network Working Group

Feng BAO
Robert DENG
Ying QIU
Jianying ZHOU

March 8, 2005

Certificate-based Binding Update Protocol (CBU)
<draft-qiu-mip6-certificated-binding-update-03.txt>

Status of this Memo

ABSTRACT

This document proposes a comprehensive security solution for mobile IPv6 networks including secure binding update, secure fast handover, user authentication and session key management for data security. In our proposal, one of the home agent's functions is to act as security proxy for its mobile nodes. The authentication is based on the home agent's certificate and the secret session keys are generated by strong cryptosystems. Our proposal avoids many security obstacles in the Return Routability protocol and provides a simple,

integrated and efficient security solution for mobile communication.

Table of Contents

1. INTRODUCTION

The demand for mobile communication requires a solution for efficient authentication of mobile nodes and protection of communications between mobile nodes. Presently, the IETF mobile working group has submitted two major drafts on mobile connection:

i) "Mobility Support in IPv6 (MIPv6)"[1], which specifies a protocol that allows mobile nodes to remain reachable while moving around in the IPv6 Internet, and
ii) "Hierarchical Mobile IPv6 Mobility Management (HMIPv6)"[2], which introduces extensions to Mobile IPv6 and IPv6 Neighbour Discovery to allow for local mobility handling.

Figure 1 illustrates the topology of communications among Mobile Node (MN), Correspondent Node(CN), Home Agent(HA) and Mobility Anchor Point (MAP). Under this architecture, the major security issue is how to authenticate MN by CN and MAP and how to protect communications in channels of MN-HA, MN-CN, HA-CN, MAP-MN and MAP-HA.

Figure 1: Topology and protected channels in MIPv6.

In the Mobile IPv6 [1], a method of Return Routability(RR) is used to process Binding Updates(BU). However, RR cannot provide satisfied level of security. Therefore, the working group suggests to bundle IKE[3] for improving the authentication ability and protecting the communication channel MN-HA.

The original goal of RR was to provide a simple way to protect the signals of binding update. But it is now used with IKE and IPsec for higher security requirement. Consequently, many patches for RR are introduced to solve the above issues, and the RR solution becomes more and more complicated. For example, in order to handle the scenario of two mobile nodes in simultaneous movement, a new RR model is introduced: one deals with fixed CN while another deals with mobile CN.

Even worse, due to the heavy computing overheads, IKE is not suitable for mobile devices (e.g., mobile phone, etc.) with very limited computing power and battery lifetime.

In this document, we propose a comprehensive security solution for mobile IPv6 networks and try to provide a simple, integrated and efficient security solution for mobile communication.

In our solution, all the security features, such as secure binding update, seamless secure fast handover, user authentication, session key management for data security, etc., are provided within one framework.

In this solution, the home agent handles strong authentication for its mobile nodes and the authentication process is mainly run between wired devices (i.e., a home agent and a wired correspondent node or the home agent of a mobile correspondent node). Moreover, the complicated session key management and security association management are also deployed on the home agent. The motivation behind this is that home agents are fixed machines with rich computing capability and are connected to wired networks with a much broader bandwidth and more stable connection than mobile networks. Therefore, our solution could keep the balance well between the strong security requirements for e-commerce and the weak capability of mobile devices in terms of computing power and communicating speed.

2. NOTATIONS

The notations used throughout this paper are listed below for easy reference:

h(): a one-way hash function, such as MD5 or SHA.
prf(k, m): a keyed pseudo random function -- often a keyed hash function. It accepts a secret key k and a message m, and generates a pseudo random output. This function is used for both message authentication and cryptographic key derivations.
e(k ,m): encryption of message m with a secret key k.
Px/Sx: a public and private key pair of node X in a digital signature scheme such as RSA or DSS.

Sig(Sx, m): node X's digital signature on a message m using a private key Sx.
m|n: concatenation of two messages m and n.

3. CERTIFICATE-BASED BINDING UPDATE PROTOCOL(CBU)

In CBU protocol, the digital signature cryptosystem is used. The public/private key pair of HA is denoted by PK_HA and SK_HA. The private key SK_HA is kept by HA in the home link, probably inside a tamper-resistant hardware of cryptogram processing device. The home link obtains a public key certificate,

Cert_HA = {HLSP, PK_HA, Valid_Interval, SIG_CA}

from a Certification Authority(CA), where HLSP is the home link subnet prefix, Valid_Interval is the valid duration of the certificate, and SIG_CA is CA's signature on HLSP, PK_HA and Valid_Interval.

Figure 2 shows message exchange between a mobile node MN, its home agent HA and its correspondent node CN in CBU protocol. The existence of and operations performed by HA are made transparent to CN.

The use of cookies during the key exchange is a weak form of protection against an attacker who generates a series of request packets, each with a different spoofed source IP address, and sends them to a protocol party. For each request, the protocol party will first validate cookies before performing computational expensive public key cryptographic operations. Below is a detailed description of the CBU protocol.

When a mobile node MN wants to start route optimization operation with a correspondent node CN, it sends a route optimization request

Figure 2. Message exchange in CBU protocol.

REQ = {Src=HoA, Des=HA, e(k_HA, HoA, CoA, CN, N0)}

to HA via IPsec-protected secure tunneling. Here, HA represents both the home agent and its IP address while CN represents both the correspondent node and its IP address. N0 is a nonce used to match the reply message REP. k_HA is a session key for the IPsec secure tunnel, its initial value is a pre-shared secret value z which is generated by HA before MN leaves its home link. How to update the session key will be described later. IPsec provides replay protection only when dynamic security association establishment is used. This may not always be possible and manual keying might be preferred in certain circumstances. For this reason, a random number N0 is included in order to counter message replay.

After decrypting the REQ message and verifying HoA, HA creates a cookie C0 and sends

COOKIE0 = {Src=HoA, Des= CN, C0}

to CN. In reply, CN generates a nonce N1 and a cookie C1, and sends

COOKIE1 = {Src=CN, Des=HoA, C0, C1, N1}

to MN. Note that the destination address in COOKIE1 is MN's home address HoA. As a result, this message is delivered to MN's home link and intercepted by HA using IPv6 Neighbor Discovery.

Upon receiving COOKIE1, HA checks on the validity of C0, generates a nonce N2 and a Diffie-Hellman secret value x < p, computes its Diffie-Hellman public value g^x and its signature

SIG_HA = Sig(SK_HA, HoA | CN | g^x | N1 | N2 | TS)

using home link's private key SK_HA, where TS is a time stamp. This time stamp does not have to be checked by the recipient during the message exchange. It will be used to trace back the culprit should a malicious mobile node flooding attack have occurred. Finally, HA replies CN with

EXCH0 = {Src=HoA, Des= CN, C0, C1, N1, N2, g^x,
TS, SIG_HA, Cert_HA},

where Cert_HA is the public key certificate of the home link as defined before. Note that the values of N1 and N2 are included in the signature SIG_HA in order to counter replay of old signatures and to resist chosen message attacks to the signature scheme, respectively.

When CN receives EXCH0, it validates the cookies, the home link's public key certificate Cert_HA, the signature and more importantly, checks for equality of the home link subnet prefix strings embedded in both Cert_HA and HoA. If all the validations and checking are positive, CN can be confident that the home address HoA of MN is authorized by its home link and the Diffie-Hellman public vaule g^x is freshly generated by MN's home link. CN next generates its Diffie-Hellman secret value y < p. It then computes its Diffie-Hellman public value g^y, the Diffie-Hellman key

k_DH = (g^x)^y,

a master secret

k_master = prf(k_DH, N1 | N2),

and three secret session keys

k_BU = prf(k_master, N1 | N2 | 0),
k_BA = prf(k_master, N1 | N2 | 1),
k_EN = prf(k_master, N1 | N2 | 2),

where k_BU is the binding key used for authenticating binding update messages from MN to CN, k_BA is the acknowledgement key used for authenticating binding acknowledge messages from CN to MN, and k_EN is the encryption key used for encrypting/decrypting packets between MN and CN. Then CN sends MN

EXCH1 = {Src=CN, Des= HoA, C0, C1, g^y, SIG_CN, Cert_CN},

where

SIG_CN = Sig(SK_CN, CN | HoA | g^y | EXCH0).

and

Cert_CN = {CN, PK_CN, Valid_Interval, SIG_CA}

is CN's certificate from CA, and CN could be CN's IP address (if CN is a wired terminal node) or CN's home link subnet prefix (if CN is a server-supported mobile node). Therefore, both parties MN and CN could identify each other, which will be useful for setting up access control on MN (or HA) and CN.

Again, this message is intercepted by HA, which first validates the cookies, calculates the Diffie-Hellman key k_DH = (g^y)^x and the master secret k_master = prf(k_DH, N1|N2). HA also generates the secret session keys

k_BU = prf(k_master, N1|N2|0),
k_BA = prf(k_master, N1|N2|1),
k_EN = prf(k_master, N1|N2|2)

and

k_HA-next = prf(z, N0 | N1),

where k_HA-next is the IPsec session key used for the IPsec-protected tunnel between MN and HA. Then HA sends

REP = {Src= CN, Des=CoA, payload}

to MN through IPsec-protected secure tunneling, where

payload = e(k_HA, N0, k_BU, k_BA, k_EN, k_HA-next).

Please note, we use the current k_HA to encrypt the next k_HA-next, which will be kept by both MN and HA for next use when MN sends a new REQ to HA.

Considering the REP message might be lost during the transfer, HA should keep the previous k_HA until it confirms MN had used the new k_HA. If MN did not receive the REP message after a reasonable interval, it will resend the REQ message. First, HA use its current k_HA(i.e. k_HA-next here) to decrypt the REQ message. If HA cannot get a HoA that is belonged to its home link subnet, it will use the previous k_HA(i.e. k_HA here) to decrypt the REQ message again. After verifying HoA, HA can simply resend the REP message to MN.

After receiving REP, MN decrypts the payload with the current k_HA and checks whether N0 is the same as the one it sent out in REQ. If so, MN proceeds to send a binding update message

BU = {Src=CoA, Des=CN, HoA, Seq#, LT_BU, MAC_BU}

to CN, where

MAC_BU = prf(k_BU, HoA | CoA | Seq# | LT_BU)

is a MAC generated with k_BU to authenticate the BU message, Seq# is a sequence number used to detect replay attack, and LT_BU is the lifetime of the binding. If BU is verified positive, CN may reply with a binding acknowledgement message

BA = {Src=CN, Des=CoA, HoA, Seq#, LT_BA, LT_EN, MAC_BA},

where Seq# is copied from the BU message, LT_BA is the granted lifetime of the binding, LT_EN is the lifetime of k_EN, and

MAC_BA = prf(k_BA, CoA | CN | Seq# | LT_BA | LT_EN)

is a MAC generated with k_BA to authenticate the BA message. Then, CN will create a binding cache entry for HoA which includes the care-of-address, session keys, lifetimes, etc. Now CN can start sending packets encrypted with k_EN to MN at CoA address.

In order to protect against the third party bombing attacks, after receiving BA, MN should reply a binding confirmation message

BC = {Src=CoA, Des=CN, HoA, Seq#, Flag, MAC_BC}

to CN, where

MAC_BC = prf(k_BA, HoA | CoA | Seq# | Flag).

is a MAC generated with k_BA to authenticate the BC message and Flag is the indicator of binding confirmation. For the consideration of performance, the CN does not need to wait the BC before shifting to the new care-of-address. If the CN can not receive the proper BC

after a certain amount of traffics, e.g 10 packets or 10 seconds, the traffic between CN and the new CoA will be stopped.

When the binding expires, which is defined by LT_BA or MN changes its domain, MN and CN can use messages BU and BA to update the binding.

Meanwhile, when k_EN expires, which is defined by LT_EN, MN sends HA a message

REQ_KEY = {Src=HoA, Des=HA, e(k_HA, HoA, CoA, CN, N0', REQKEN)}

where REQKEN indicates to update k_EN. After decrypting the REQ_KEY message and verifying HoA, HA generates a new nonce N1' and sends

EXCH0' = {Src=HoA, Des=CN, N1', MAC_EXCH0'}

to CN, where

MAC_EXCH0' = prf(k_EN, HoA | N1').

If EXCH0' is verified positive, CN also generates a new nonce N2' and replies

EXCH1' = {Src=CN, Des=HoA, N1', N2', LT_EN, MAC_EXCH1'}

to HA, where

MAC_EXCH1' = prf(k_EN, CN | N1' | N2' | LT_EN);

After verifying and getting the new nonces N0' and N1', both HA and CN compute the new encryption key respectively

k_EN-new = prf(k_master, N1' | N2' | 2),

then HA forwards the new encryption key

NEWKEY = {Src=CN, Des=CoA, e(k_HA, N0', k_EN-new, LT_EN)}.

to MN for following packets between MN and CN .

As the messages of REQ_KEY and NEWKEY might get lost during the transfer, MN could resend the requst message REQ_KEY if it does not receive the NEWKEY message after a resonable interval.

In this proposal, messages for a new binding update request (referring to Figure 2.) are long-term, i.e., throughout a communication session or even across multiple sessions, while messages for the follow-up binding updates and the IPsec session keys are short-term, which are decided by their lifetimes LT_BA and LT_EN, respectively. In most of time, when MN changes its CoA, it only needs to send the BU message to CN. Therefore, our protocol

could get high performance in the handover process.

In the CBU protocol, as described above, the security association (binding cache entry) is based on the addresses of CN and CoA. Therefore, the IPsec could be used as usual. On the contrast, in RR bundled IKE method, the security association is based on addresses of CN and HoA, the IPsec cannot be used without modification because HoA is not deployed as source/destination address in the header of IP packets.

A major feature of our CBU protocol is provision of key management in the IKE style for protecting the channels of MN-HA, MN-CN and HA-CN.

4. SECURE BINDING UPDATE BETWEEN TWO MOVING MOBILE NODES

In this section, we discuss how to use CBU protocol in the scenarios of two mobile nodes in simultaneous movement.

In Figure 3, CN_MN is a mobile correspondent node with home address CN_HoA and care-of-address CN_CoA while CN_HA is its home agent. The messages between NM, HA and CN_HA are the same as those shown in Figure 2 but the CN's address is replaced by CN_HoA.

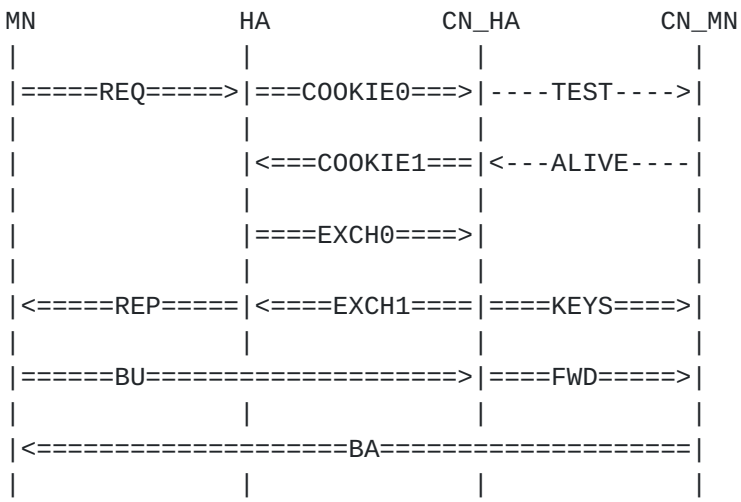

Figure 3. Message exchange between two mobile nodes.

Messages TEST and ALIVE are optional for testing whether CN_MN is alive.

TEST = {Src=CN_HA, Des=CN_CoA, test_flag}
ALIVE = {Src= CN_CoA, Des=CN_HA, N3}

where N3 is a nonce generated by CN_MN.

After computing secret session keys k_BU, k_BA and k_EN, CN_HA also sends these keys

KEYS = {Src=CN_HA, Des=CN_CoA, e(k_CN, N3, k_BU, k_BA, k_EN)}

to CN_MN through their own IPsec-protected tunnel, encrypted with their preset IPsec session key k_CN.

When CN_HA intercepts the binding message BU from MN, CN_HA simply forwards the message to CN_MN through the reserved tunnel

FWD = {Src=CoA, Des= CN_CoA, HoA, Seq#, LT_BU, MAC_BU}.

Upon receiving FWD and checking the validity of MAC_BU, CN_MN Returns

BA = {Src=CN_CoA, Des=CoA, CN_HoA, Seq#, LT_BA, LT_EN, MAC_BA}

to MN in order to inform MN of its current care-of-address CN_CoA. MAC_BA is calculated as

MAC_BA = prf(k_BA, CoA | CN_CoA | Seq# | LT_BA | LT_EN).

After receiving the messages of BU and BA, MN and CN_MN will create a binding cache entry for each other, respectively, which includes care-of-addresses of the both peers, session keys, lifetimes, etc. Then the two mobile nodes can start sending packets encrypted with k_EN to each other at their CoA addresses.

As described above, the home agent always negotiates with a fixed peer, either a fixed CN or the home agent (CN_HA) of a mobile CN. When the initial mobile node (MN) sends its current CoA in the BU message to the correspondent home agent (CN_HA), CN_HA will forward MN's CoA to its mobile node (CN_MN), and CN_MN will send its current CoA to MN directly. Therefore, from MN's point of view, it never takes care whether the correspondent node is a moving node or not. On the contrary, in the RR protocol, a special message type is used to process the scenario of two mobile nodes in simultaneous movement.

5. IMPLEMENTATION of CBU in HMIPv6

In the protocol of HMIPv6 [2], the concept of Mobility Anchor Point (MAP) is introduced. MAP is a router located in a domain visited by the mobile nodes. MAP provides the localized mobility management for the visiting mobile nodes.

Every mobile node bundles three addresses: Home Address (HoA), Regional Care-of-Address (RCoA), and On-Link Care-of-Address (LCoA). RCoA is an address on the MAP subnet, and obtained by the mobile Node (MN) from the visited domain. LCoA is configured on a MN's interface based on the prefix advertised by its default router. In fact, it is a care-of-address in normal MIPv6. Figure 4 shows the architecture of HMIPv6.

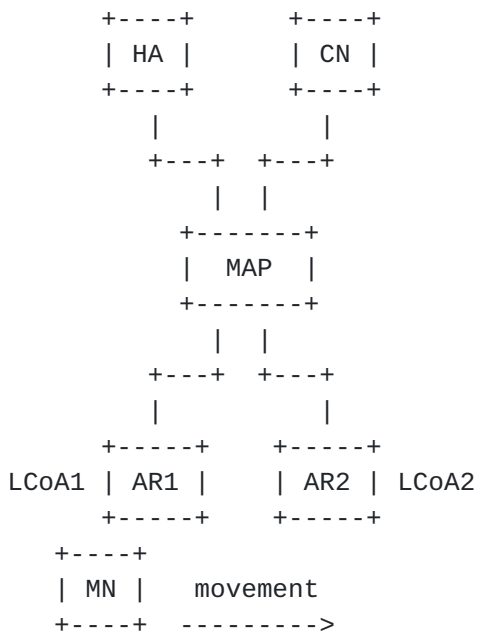

Figure 4. Hierarchical MIPv6 domain.

In HMIPv6, when CN sends packets to MN's RCoA, MAP intercepts the packets and forwards the packets to MN's LCoA. However, as the binding update message from MN to MAP is not authenticated when MN changes its Access Router (AR), attackers can easily launch "Redirect Attacks", i.e., the attacks which redirect the traffic from MAP to fake destinations chosen by the attackers. Therefore, in this section, we propose an approach to protect the binding update message from MN to MAP.

When MN roams in the MAP domain, as soon as MN attaches to another AR and gets a new LCoA, MN will send a BU message to MAP with its certificate and signature

BU = { Src=LCoA, Des=MAP, HoA, RCoA, SIG_MN, Cert_MN }

where

SIG_MN = Sig{SK_MN, LCoA | MAP | HoA | RCoA},

and

Cert_MN = {HoA, PK_MN, Valid_Iinterval, SIG_HA}.

SK_MN and PK_MN are a private and public key pair for MN. MN's public key certificate is issued by its home agent (HA). Here, SIG_HA is HA's signature on HoA, PK_MN and Valid_ Interval.

After MAP got the message, if MAP does not have HA's public key certificate, MAP will send a request for certificate to HA,

REQ_Cert = {Src=MAP, Des=HA, request_cert}.

Then, HA will return to MAP its public key certificate issued by a CA,

REP_Cert = {Src=HA, Des=MAP, Cert_HA}.

Upon getting HA's public key certificate, MAP can verify HA's signature SIG_HA and trust the MN's public key certificate Cert_MN. With Cert_MN, MAP can further verify MN's signature. After double checking the equality of home link's subnet prefix string embedded in both Cert_HA and HoA, MAP can finally trust MN's new binding update message BU.

6. SECURITY CONSIDERATIONS

The proposal of Certificate-based Binding Update (CBU) solves many security issues in mobile networks, i.e. the secure binding update, seamless secure fast handover, user authentication, session key management for data security, etc.

With the use of a digital signature scheme and the Diffie-Hellman key exchange algorithm, the CBU protocol can prevent the Session Hijacking attacks (an intruder hijacks an existing session between a mobile node and a correspondent node and redirects the correspondent node's traffic to a malicious location) and the Malicious Mobile Node Flooding attacks (a mischievous mobile node sets up communication sessions with correspondent nodes, and then redirect traffic from the correspondent nodes to flood a victim node or network). For the detail analysis, please refer to [4].

7. CONCLUSION

As described in this document, the CBU protocol provides a key management scheme to protect all the channels, e.g., the channels of MN-HA, MN-CN, HA-CN as well as MN-MAP, HA-MAP in mobile IPv6 networks. For the channel MN-HA, because HA acts as a security agent for MN and they share a secret value, our scheme provides a dynamic encryption key for this IPsec channel. For the channel MN-CN, our scheme manages both the long-term security associations for authentication and the short-term security associations for packet encryption in the channel. For the channel HA-CN, authentication is also provided.

Thanks to a strong cryptosystem used in the proposal, in most of time, MN can simply send the authenticated binding update messages (BU) to its CN when it enters into other subnets, without re-establishing new authentication keys via HA. So it is more suitable for fast handover in mobile networks.

Due to the dynamic security associations (SA) for IPsec channels based on CoAs of these two nodes, our scheme can avoid the indexical problem that RR protocol must face, in which IP addresses in IP packets do not match the IP addresses in the SA database.

As both CoAs of two nodes are never involved in the process of security negotiation in our scheme, either fixed CN or mobile CN can be dealt with the same method. Contrastively, RR protocol has to provide different methods to process these two scenarios.

Because the major security negotiation in our scheme is carried out on the fixed machines (home agents or fixed CNs) connected with the wired networks, our solution can significantly reduce the computing and communication requirements on the mobile nodes.

Therefore, our proposal could keep well a balance between the strong security requirements for e-commerce and the weak capability of mobile devices, and provides a more tidy, integrated, practical and efficient security solution for mobile networks.

8. ACKNOWLEDGEMENTS

The authors would like to thank James Kempf for his valuable comments and suggestions.

9. REFERENCES

[1] D. B. Johson and C. Perkins, "Mobility Support in IPv6", IETF INTERNET-DRAFT, July 2003.
[2] H. Soliman and K. El-Malki, "Hierarchical MIPv6 Mobility Management (HMIPv6)", July 2003.
[3] D. Harkins and D. Carrel, "The Internet Key Exchange (IKE)", RFC 2409, November 1998.
[4] R. Deng, J. Zhou and F. Bao, "Defending against Redirect Attacks in Mobile IP", Proceedings of 9th ACM Conference on Computer and Communications Security, pages 59--67, Washington, DC, November 2002, ACM Press.

10. AUTHORS' ADDRESSES

Feng BAO
Institute for Infocomm Research
21 Heng Mui Keng Terrace
Singapore 119613
Phone: +65-6874-8456
EMail: baofeng@i2r.a-star.edu.sg

Robert DENG
Singapore Management University
469 Bukit Timah Road
Singapore 259756
Phone: +65-6822-0920
EMail: robertdeng@smu.edu.sg

Ying QIU
Institute for Infocomm Research
21 Heng Mui Keng Terrace
Singapore 119613
Phone: +65-6874-6742
EMail: qiuying@i2r.a-star.edu.sg

Jianying ZHOU
Institute for Infocomm Research
21 Heng Mui Keng Terrace
Singapore 119613
Phone: +65-6874-8543
EMail: jyzhou@i2r.a-star.edu.sg

A. CHANGE LOG

The following changes have taken place since the previous version.

- In section 3, add a binding confirmation message (BC) in order to protect against the third party bombing attacks.

- Revision of IPSec to IPsec.

FOR A PARTICULAR PURPOSE.

COPYRIGHT STATEMENT

ACKNOWLEDGMENT

Funding for the RFC Editor function is currently provided by the Internet Society.