LETTER

A Security Analysis on Kempf-Koodli's Security Scheme for Fast Mobile IPv6

Ilsun YOU[†a)], **Kouichi SAKURAI**[††], ***and*** **Yoshiaki HORI**[††], ***Members***

SUMMARY Recently, the security scheme, proposed by Kempf and Koodli, has been adopted as a security standard for Fast handover for Mobile IPv6. But, it does not prevent denial of service attacks while resulting in high computation cost. More importantly, we find that it is still vulnerable to redirection attacks because it fails to secure the *Unsolicited Neighbor Advertisement* messages. In this paper, Kempf-Koodli's scheme is formally analyzed through BAN-logic and its weaknesses are demonstrated.

key words: *Fast Mobile IPv6 security, SEND protocol, CGA, BAN-logic*

1. Introduction

By exploiting various L2 triggers, Fast Mobile IPv6 (FMIPv6) addresses the excessive latency caused during the mobile node's handover [1]. Without any security mechanism, it is susceptible to various security threats such as redirection attacks. Several approaches have been conducted to protect FMIPv6 [2]–[5]. They attempt to securely distribute a handover key between a mobile node and an access router by relying on the SEcure Neighbor Discovery (SEND) protocol [6] or the Authentication, Authorization, and Accounting (AAA) infrastructure [7]. Among them, the security scheme of Kempf and Koodli has been adopted as a standard (RFC 5269) [2]. It achieves the most secure handover key distribution by depending upon the SEND protocol which based on the Cryptographically Generated Address (CGA) method [8]. However, the SEND protocol causes the scheme to suffer from denial of service attacks and high computation cost. More importantly, we discover that it is still vulnerable to redirection attacks since it does not protect the *Unsolicited Neighbor Advertisement* (*UNA*) message. In this paper, we show Kempf-Koodli's scheme is not correct through BAN-logic [9], which is widely used for the formal analysis of security protocols, and then demonstrate its weaknesses.

2. Review of Kempf-Koodli's Scheme

2.1 Notations and Preliminary

- $Msg(address1, address2)$ means that the message Msg is sent from $address1$ to $address2$.
- $E(K, M)$ denotes a function that encrypts the message M with the given key K, where K can be a secret key or a public key.
- $SIGN(K, M)$ denotes a function that digitally signs the message M with the private key K.
- $AR(i)$ denotes the ith access router which the MN visits, and its IPv6 address, where $i > 0$.
- $CoA(i)$ is the ith care-of address of the MN, where $i > 0$.
- PU_X is the X's public key from which the CGA is derived.
- PR_X is the X's private key which corresponds to PU_X.
- $HK(i)$ is the ith handover key, where $i > 0$.
- $CGA - PARAM_X$ is the X's parameters which are used to verify that the X's CGA is derived from PU_X.
- AT is an algorithm type describing the algorithm used to calculate the authenticator of the FBU message.
- $HEPK_{MN}$ is the MN's public key which is used to encrypt a handover key.
- $HERK_{MN}$ is the MN's private key which is used to decrypt a handover key.

It is assumed that each entity has a public/private key pair (PU_{MN}/PR_{MN} or $PU_{AR(i)}/PR_{AR(i)}$) and its address is a CGA. It is also assumed that each mobile node possesses a handover encryption public/private key pair ($HEPK_{MN}/HERK_{MN}$), which is generated using the same public key algorithm as for the SEND protocol. Note that the key pair should be used for only handover negotiation. Moreover, we assume that there is a secure channel between access routers.

2.2 Operation

Figure 1 shows Kempf-Koodli's scheme, which is composed of three phases: handover key negotiation, fast binding update and new network attachment phases.

In order to protect FMIPv6, the scheme adopts the SEND protocol, which is based on the CGA method. Through the CGA method, the MN and the $AR(i)$ can verity that each other's address and public key are valid, thus safely utilizing public key cryptography to protect signaling messages as well as negotiate a handover key with each other.

(1) Handover key negotiation phase: During this phase, the MN negotiates a handover key with the current access router $AR(i)$ through its handover encryption public key

Manuscript received August 27, 2008.
Manuscript revised December 26, 2008.

[†]The author is with School of Information Science, Korean Bible University, South Korea.
[††]The authors are with the Dept. of Computer Science and Communication Engineering, Kyushu University, Fukuoka-shi, 819-0395 Japan.
a) E-mail: isyou@bible.ac.kr

DOI: 10.1587/transcom.E92.B.2287

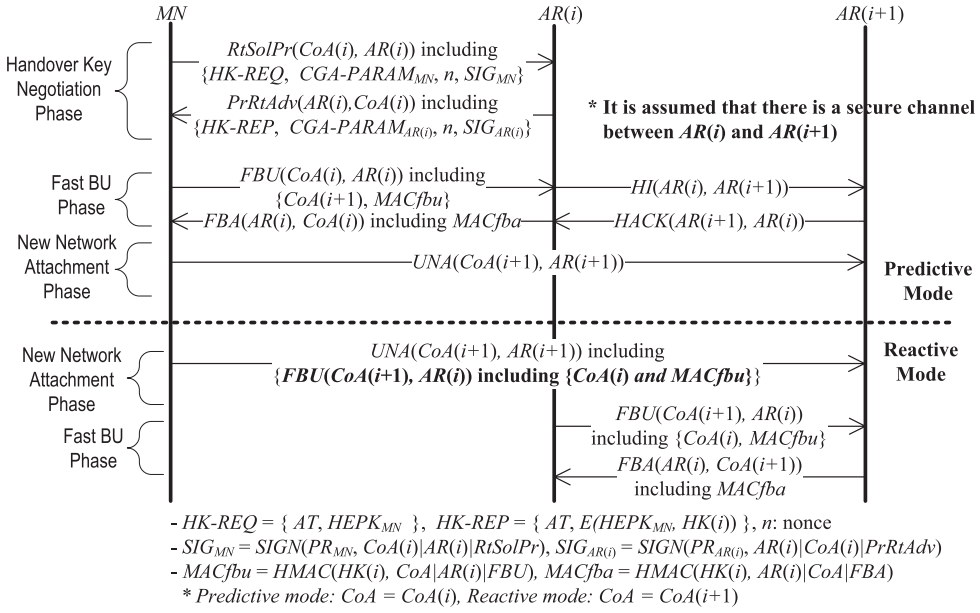

Fig. 1 Kempf-Koodli's scheme.

$HEPK_{MN}$. For this goal, the scheme exploits the existing *Router Solicitation for Proxy Advertisement* (*RtSolPr*) and *Proxy Router Advertisement* (*PrRtAdv*) messages. If the *MN* needs to execute the *FBU* protocol with the *AR*(*i*) in the future, the *MN* can reuse that handover key instead of generating a new handover key. For such a case, the *AR* stores the shared handover key with the *MN*'s CGA, which identifies that key.

(2) Fast Binding Update and New Network Attachment phases: Once the shared handover key $HK(i)$ has been constructed, the *MN* sends the *Fast Binding Update* (*FBU*) message to the *AR*(*i*) when a link-specific handover event occurs. In order to verify the *FBU* message, the *AR*(*i*) first finds $HK(i)$, which corresponds to $CoA(i)$ included in that message. If $HK(i)$ is discovered, the *AR*(*i*) makes use of it to verify the *FBU* message, and then exchanges the *Handover Initiate* (*HI*) and *Handover Acknowledge* (*HACK*) messages with the next access router $AR(i+1)$. After that, the *AR*(*i*) returns the *MN* the *Fast Binding Acknowledge* (*FBA*) message containing the authenticator $MACfba$ while stating to tunnels the traffic sent to $CoA(i)$ on its link to $CoA(i+1)$ on the $AR(i+1)$'s link. If the *FBA* message is valid, the *MN* assumes that data packets are being forwarded to its new location. As soon as the *MN* handovers to the $AR(i+1)$'s link, it announces its attachment by sending the *UNA* message to the $AR(i+1)$. When the *MN* cannot send the *FBU* message or receive the *FBA* message on the *AR*(*i*)'s link (the reactive mode), it encapsulates the *FBU* message in the *UNA* message.

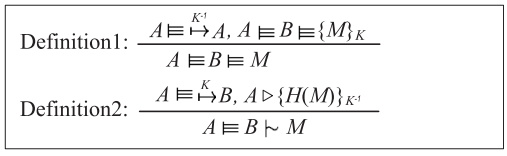

Fig. 2 Definitions.

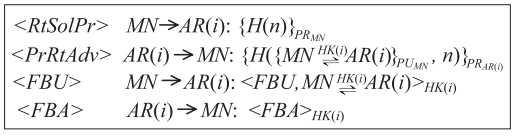

Fig. 3 Idealized form.

3. Formal Verification

In this section, we use BAN-logic to verify the correctness of Kempf-Koodli's scheme. BAN-logic was introduced by Burrows, Abadi and Needham in 1989 [9]. Since then, due to simplicity and robustness, it has become the best-known and most influential method for formally analyzing and verifying security protocols. In BAN-logic, the verification of a protocol is typically performed as follows: (*i*) idealize the original protocol, (*ii*) define assumptions about the initial state and (*iii*) apply logical postulates repeatedly until getting the intended results. For details on notations and logical postulates of BAN-logic, refer to [9].

In order to verify Kempf-Koodli's scheme, we first make two definitions as shown in Fig. 2. It is clear from the meaning of the definitions that they are intuitively true. The idealized form of Kempf-Koodli's scheme is described in Fig. 3. In the form, the $HMAC(K, M)$ operation is expressed as $< M >_K$. Especially, because the *MN* can send the *AR*(*i*)

A1: $MN \equiv AR(i) \mapsto MN \overset{HK(i)}{\rightleftharpoons} AR(i)$
A2: $MN \equiv \overset{rMN}{\mapsto} MN$
A3: $MN \equiv \overset{uMN}{\mapsto} MN$
A4: $MN \equiv \overset{uAR}{\mapsto} AR(i)$
A5: $MN \equiv \#(n)$
A6: $MN \equiv \#(seq)$
A7: $MN \equiv AR(i) \mapsto FBA$
A8: $AR(i) \equiv MN \overset{HK(i)}{\rightleftharpoons} AR(i)$
A9: $AR(i) \equiv \#(MN \overset{HK(i)}{\rightleftharpoons} AR(i))$
A10: $AR(i) \equiv \overset{uMN}{\mapsto} MN$
A11: $AR(i) \equiv \overset{rAR}{\mapsto} AR(i)$
A12: $AR(i) \equiv \overset{uAR}{\mapsto} AR(i)$
A13: $AR(i) \equiv MN \mapsto FBU$
* $rMN = PR_{MN}$, $uMN = PU_{MN}$, $rAR = PR_{AR(i)}$, $uAR = PU_{AR(i)}$,

Fig. 4 Idealized form.

From the *RtSolPr* message, we derive:
(1) $AR(i) \equiv MN \sim n$ {by Definition2, A10}
From the *RPrRtAdv* message, we derive:
(2) $MN \equiv AR(i) \sim (\{MN \overset{HK(i)}{\rightleftharpoons} AR(i)\}_{PU_{MN}}, n)$ {by A4, Definition2}
(3) $MN \equiv \#(\{MN \overset{HK(i)}{\rightleftharpoons} AR(i)\}_{PU_{MN}}, n)$ {by A5}
(4) $MN \equiv AR(i) \equiv \{MN \overset{HK(i)}{\rightleftharpoons} AR(i)\}_{PU_{MN}}$ {by (2), (3), R2}
(5) $MN \equiv MN \overset{HK(i)}{\rightleftharpoons} AR(i)$ {by A2, (4), Definition1, A1, R3}
From the *FBU* message, we derive:
(6) $AR(i) \equiv MN \sim (FBU, MN \overset{HK(i)}{\rightleftharpoons} AR(i))$ {by A8, R1}
(7) $AR(i) \equiv \#(FBU, MN \overset{HK(i)}{\rightleftharpoons} AR(i))$ {by A9}
(8) $AR(i) \equiv FBU$ {by (6), (7), R2, A13, R3}
From the *FBA* message, we derive:
(9) $MN \equiv FBA$ {by (5), R1, A6, R2, A7, R3}
* R1 denotes the message-meaning rule, R2 denotes the nonce verification rule, R3 denotes the jurisdiction rule

Fig. 5 Kempf-Koodli's scheme.

the valid FBU message only if it knows $HK(i)$, the second part is included in the FBU message of the idealized form. The assumptions are defined in Fig. 4. While $A5$-$A7$ and $A13$ are about the freshness of the nonces and the authority on the messages, other assumptions are about the shared handover key and the private key/public key pairs. Strictly speaking, in BAN-logic, we cannot prove that PU_{MN} belongs to the MN as well as $PU_{AR(i)}$ belongs to the $AR(i)$ though they can be verified by the CGA method. Thus, we present $A4$ and $A10$ instead. With the idealized form, definitions and assumptions, we can verify Kempf-Koodli's scheme as described in Fig. 5. First, the verification focuses on checking if the handover key negotiation is correct during steps (1)–(5). After that, the FBU and FBA messages are validated during steps (6)–(9). In (9), because the FBA message includes seq, the MN can believe that this message is fresh and consequently true. From (8), we can know that the $AR(i)$ believes the FBU message. Based on such belief, it acts as a temporary home agent while redirecting the MN's traffic to the $AR(i+1)$. However, because the $AR(i+1)$ just sees the UNA message, it is not sure that the MN truly arrives at its link. The unbelievable UNA message causes the scheme to be vulnerable to several attacks. As a result, we can conclude that the scheme is not correct.

4. Weaknesses of Kempf-Koodli's Scheme

As shown above, Kempf-Koodli's scheme does not protect the UNA message while focusing on only securing the handover negotiation phase. Moreover, it suffers from high computation cost and denial of service attacks. In this section, we analyze such weaknesses of the scheme in detail.

4.1 Insecure UNA Messages

With insecure UNA messages, an attacker can launch the following redirection attacks.

- Session hijacking attacks based on eavesdropping: It is assumed that the attacker can eavesdrop all traffic between the victim MN and the $AR(i)$, and thus know the MN's handover context. If the attacker captures the FBU message from the MN, it tries to attach the link of the $AR(i+1)$ with the MN's $CoA(i+1)$ and then send the fabricated UNA message to that router faster than the MN. If such a trial is successful, the $AR(i+1)$ cannot detect if the UNA message is forged. As a result, it starts to forward the traffic sent to the MN to the attacker.
- Session hijacking attacks based on the stolen handover key: It is assumed that the attacker somehow succeeds in stealing the victim MN's current handover key $HK(i)$. In this case, the attacker can easily make use of the reactive mode in order to launch the session hijacking attack. Note that if the predictive mode is used, the victim node can detect that the other node tries to masquerade itself by receiving the unexpected FBA message. After the attacker moves to another link, it sends that link's access router $AR(i+1)$ the UNA message including the fabricated FBU message, whose authenticator is generated with the stolen handover key. After processing the UNA message, the $AR(i+1)$ performs the fast binding update phase with the MN's current access router $AR(i)$. As a result, the traffic toward the MN is redirected to the attacker.
- Malicious flooding attacks: The purpose of this attack is to cause the target network to be flooded with unwanted excess traffic. This attack is executed by two attackers: $Attacker1$ and $Attacker2$. It is assumed that two attackers can communicate with each other. While the $Attacker1$ is a legitimate mobile node and initiates the attack, the $Attacker2$ exists in the target network and sends a forged UNA message to let this attack continued. In order to initiate this attack, the $Attacker1$ first makes sessions with corresponding nodes, which can result in excess traffics. Then, it sends an FBU message indicating that it will move to the target network. Note that at this point the $AR(i)$ starts to tunnel the $Attacker1$'s traffic to the target network's access router $AR(i+1)$. The lifetime or the buffer size for the established tunnel is limited until the $AR(i+1)$ receives the $Attacker1$'s UNA message. Instead of handover, the $Attacker1$ just requests the $Attacker2$ to send an UNA message indicating that the $Attacker1$ attaches the target network. The $Attacker2$ transmits the requested message to the $AR(i+1)$, which then release the

Table 1 Computation cost of Kempf-Koodli's scheme (S: the cost for the one signature generation, V: the cost for verifying the one signature, H: the cost for one hash operation, HM: the cost for one HMAC operation, E: the cost for one public key encryption, D: the cost for one public key decryption).

Message	*MN*	*AR*
RtSolPr	S	2H+V
PrRtAdv	2H+V+D	E+S
FBU	HM	HM
FBA	HM	HM
UNA	0	0
Total	S+V+D+2H+2HM	S+V+E+2H+2HM
*Values	86,035 K	46,061 K

* The parameters provided in [10] are applied to compute the values. It is assumed that the size of the message is 100 byte message and the used algorithms are SHA1, HMAC-SHA1 and RSA. The values are measured in cycles.

limitation on the tunnel established for the *Attacker*1. As a result, the *Attacker*1's traffic is redirected to the target network.

4.2 High Computation Cost

As shown in Table 1, the *MN* and the *AR*(i) should perform at least three asymmetric cryptographic operations. Such a computation overhead imposes considerable burdens on mobile nodes, which generally tend to have limited computational capabilities and low battery power.

4.3 Denial of Service Attacks

Because of being protected by the SEND protocol, the digitally signed *RtSolPr* and *PrRtAdv* messages cause the *AR*(i) to be vulnerable to the denial of service (DoS) attack. In addition, because more than one CGA can be derived from one public key, one attacker is able to masquerade as many *MN*s. Especially, if the *AR*(i) accepts that the Sec value is 0, the attacker can easily and efficiently generate CGAs by using just one public/private key pair. Even more, she or he can perform the DoS attack while making CGAs in real time. The attacker launches the DoS attack as below:

(a) Generates a private/public key pair.
(b) Derives N CGAs from the public key.
(c) Uses the private key and the generated CGAs to make, sign and send N *RtSolPr* messages.

5. Conclusion

This paper analyzed the weaknesses of Kempf-Koodli's scheme. The formal verification, based on BAN-logic, showed that it is not correct due to insecure *UNA* messages. Especially, it was demonstrated that this drawback causes the scheme to be still vulnerable to several redirection attacks. In addition, we analyzed the scheme's computational cost and how to launch the DoS attack.

Acknowledgment

This work is supported by the JSPS RONPAKU program.

References

[1] R. Koodli, "Fast handovers for mobile IPv6," IETF RFC 5268, July 2005.
[2] J. Kempf and R. Koodli, "Distributing a symmetric FMIPv6 handover key using SEND," IETF RFC 5269, June 2007.
[3] V. Narayanan, N. Venkitaraman, H. Tschofenig, G. Giaretta, and J. Bournelle, "Establishing handover keys using shared keys," IETF draft-vidya-mipshop-handover-keys-aaa-04, March 2007.
[4] J. Choi and S. Jung, "A secure and efficient handover authentication based on light-weight diffie-hellman on mobile node in FMIPv6," IEICE Trans. Commun., vol.E91-B, no.2, pp.605–608, Feb. 2008.
[5] W. Haddad and S. Krishnan, "Authenticating FMIPv6 handovers," IETF draft-haddad-mipshop-fmipv6-auth-02, Sept. 2006.
[6] J. Arkko, J. Kempf, B. Zill, and P. Nikander, "SEcure neighbor discovery (SEND)," IETF RFC 3971, March 2005.
[7] C. Perkins and P. Calhoun, "Authentication, authorization, and accounting (AAA) registration keys for mobile IPv4," IETF RFC 3957, March 2005.
[8] T. Aura, "Cryptographically generated addresses (CGA)," IETF RFC 3972, March 2005.
[9] M. Burrows, M. Abadi, and R. Needham, "A logic of authentication," ACM Trans. Comput. Syst., vol.8, no.1, pp.18–36, 1990.
[10] B. Preneel, et al., "Performance of optimized implementations of the NESSIE primitives," Technical Report, NES/DOC/TEC/WP6/D21/2, Feb. 2003.