Secure Connection in Mobile IPv6

Hero Modares[1], Amirhossein Moravejosharieh[2], Rosli Salleh[3]
[1,3]Department of Computer system and technology, University of Malaya, Kuala Lumpur, Malaysia
[2]Department of Computer, Science and Engineering, University of Canterbury, Christchurch, New Zealand
Hero.Modares@gmail.com

Abstract: The MIPv6 protocol allows a Mobile Node (*MN*) to maintain its network connection during attachment transfers. For Mobile IPv6 to run properly it is important to keep Binding Update (*BU*) messages protected. The *BU* message contains sensitive data. It consists of a Mobile Node's Home Address (*MN's HoA*) and Care-of Address (*CoA*). It is responsible for redirect traffic among addresses. These messages suffer from vulnerabilities in security that gives attackers the opportunity to redirect traffic for interception by sending false binding update messages . This can cause the breach of user privacy and modifying of data packets being transferred for the benefit of the attackers. There have been many security solutions introduced to protect these messages from weaknesses that could be used by attackers but security is still an issue when it comes to accessing information anywhere and anytime. In this work our main interest is to protect binding update messages which are sent from the mobile node to the corresponding node.

[Hero Modares, Amirhossein Moravejosharieh, Rosli Salleh. **Secure Connection in Mobile IPv6.** *Life Sci J* 2013;10(2):1663-1667]. (ISSN:1097-8135). http://www.lifesciencesite.com. 232

Keywords: Mobile IPv6, Security in Mobile IPv6, Binding Update Message

1. Introduction

IP mobility is a protocol for mobile networking and computing, and from the convergence of the Internet and mobile communication technologies. IP mobility is designed to allow a Mobile Node (*MN*) to move from one network to another during communication, even though the *MN*'s point of attachment to the network has physically changed (Perkins 1997). When a mobile device is disconnected from the present network and gets reconnected to another network, portability is achieved. The *MN* achieves portability by having two IP addresses:

- **The Home Address (*HoA*):** a static IP address that resides in the home network.
- **The Care-of Address (*CoA*):** the *MN's* address in its foreign network.

Whenever a Correspondent Node (*CN*) attempts to communicate with the *MN* located in a foreign network, it sends a packet to the Home Agent (*HA*). The *HA* will use IP encapsulation, whereby an outer header is added to the data packet along with the new address, and it then proceeds to tunnel the packet to the *MN*'s *CoA* at which point the packet will be encapsulated (Figure 1).

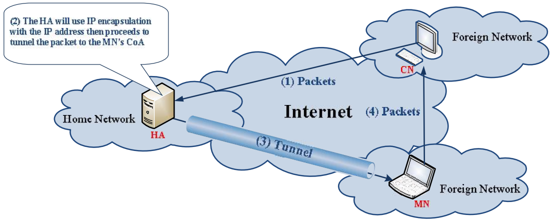
(2) The HA will use IP encapsulation with the IP address then proceeds to tunnel the packet to the MN's CoA
Foreign Network
CN
(1) Packets
Home Network
HA
Internet
(4) Packets
(3) Tunnel
Foreign Network
MN

Figure 1:Packet Exchange Between the Nodes in IP Mobility Protocol

In the IP Mobility, Binding Update (*BU*) message is sent from the *MN* to the *HA* and *CN*. This *BU* message is very important as it contains the Home Address (*HoA*) and Care-of Address (*CoA*). There are several vulnerability issues with *BU* messages, and these include: data packet interception where an attacker is able to eavesdrop on the content causing a breach in confidentiality; modification of the transmitting packets to suit the malicious purpose of the attacker; Men-In-The-Middle (MITM) attacks; session hijacking attacks; Denial of Service (DoS) attacks and Return to Home spoofing attacks. In order for the attacks to happen, the attacker must know the IPv6 addresses of the *MN*'s *HoA* and the *CN* (Fu, Lin et al. 2011; Arshad, Farooq et al. 2012).

2. Problem Statements

The IPv6 has a number of security vulnerabilities particularly the vulnerability of the *BU* message to various attacks. In the IPv6 the *MN* needs to authenticate itself every time it moves to a foreign network. Without a secure authentication process, the data packets sent from one node to another node might be intercepted by a malicious node, causing the data packets to be redirected to its location or to an arbitrary IP address. This will deny service for the intended legitimate receiver node. This happens because current protocols do not have an effective authentication method to check the validity of the users or to conceal the *HoA* and *CoA* of the *MN* (Georgiades 2011). It is safe to conclude that most protocols face security vulnerabilities due to the because they do not have an ineffective way of:

- authenticating a user's authority, hence, this is no way for the *CN* to know that the *MN* is a valid node and not a malicious one.
- hiding the Care-of Address (*CoA*) in the current MIPv6 security protocols, hence, there is no way for the *CN* to validate the owner of the address, especially, *CoA*s with spoofed or modified address.

In view of the weaknesses of the protocols, mentioned above, effective measures must be taken to provide a secure connection between the *MN* and the *CN* to protect the *BU* messages from common attacks on the MIPv6 protocol. These attacks include: Denial of Service attacks, where the attacker impersonates and denies services intended for the target node by directing network traffic to itself or another node (false node). Men-In-The-Middle attacks; and session hijacking attacks.

3. Binding Update Protection

There are two types of algorithm in cryptography: symmetric, and asymmetric. In symmetric algorithms, the encryption and decryption processes use the same key. It is easy to implement cryptography, but the difficulty is in maintaining data securing and secrecy during data exchange. Many security solutions such as the Diffie-Hellman algorithm and the traditional RSA public key use cryptography to provide security for *BU* messages (Xiao 2003). The Diffie-Hellman algorithm communicates using a key (Boyko, MacKenzie et al. 2000). A major drawback, however, is that the Diffie-Hellman algorithm cannot determine the validity of the keys sent by the sender or the receiver (Werapun and Unakul 2004).

Asymmetric cryptography helps to mitigate the problems encountered in the use of the symmetric algorithm. The asymmetric algorithms, two keys are used (a private key and a public key). The private key is only available to the owner; and the public key is available to everyone. This type of algorithm runs complex mathematical formulae with exponentiation and large prime numbers (Rivest, Shamir et al. 1978). In order for someone to send a message to another person, he must first obtain the receiver's public key to encrypt the message. The encrypted message is then forwarded to the recipient where it is to be decrypted with the recipient's private key to obtain the plaintext. Traditional public key cryptography is applied to the RSA method. This security model is popular in various networks including MIPv6. The model, however, consumes a large amount of bandwidth owing because of the recommended RSA key length of 1024 bits to improve security. The security levels of an Elliptic Curve Cryptography (ECC)-based system with a 160-bit prime order is similar to the 1024-bit modulus p DSA model, and the 1024-bit modulus n RSA model (Menezes 1993) (Boneh and Franklin 2003). In this research, the Elliptic Curve Cryptographic system (Miller 1986) and the Elliptic Curve Digital Signature Algorithm (ECDSA) (Werapun and Unakul 2004).

There are a variety of authentication architectures in existence today such as Kerberos (Kohl and Neuman 1993) but they mostly refer to a central authentication database (infrastructure-based) for user verification. An infrastructure-less or a decentralised authentication system would provide a much better security for mobile IP. A commonality can be obtained by understanding all the fundamental components of many different type authentication systems. This commonality can be used to create the final proposed solution. They will include components for digital signatures (Tanenbaum and Van Steen 2003), hashes (Arkko, Nikander et al. 2002), address based keys (Kempf, Gentry et al. 2002) and address that are cryptographically generated (Aura 2005).

4. Proposed framework

Our proposed method to assert the address ownership of the *MN* is reliant on the *MN* home link's assistance. It requires the home link to enable the *CN* to securely verify the authenticity of the *MN's HoA* ownership. A 128-bit IPv6 address is used in this method and is given at 64-bit subnet prefix with 64-bit identifier. The strong cryptographic binding between the *MN* and the *CN* provided by this method uses the interface identifier. It allows the *MN* to have its own private and public keys. The PKI is not needed anymore with this binding ownership of the *MN*'s *HoA*. By sending the 128-bit IPv6 address hash, the home address proof is therefore implemented. It can be done in two different ways. Initially, the *MN* should retrieve its private and public key pair. The user's ID is used to obtain the node's private and public key pairs. The user ID we will use is the node's Media Access Control address (MAC).

In the trust method, the responsibility of ensuring the binding correctness of the *MN* and its *HoA* to the correspondent node falls to the *MN*. Therefore, the *CN* must be assured by the *MN* that:

- The *HoA* ownership belongs to the *MN*.
- The *MN* is the actual *BU* request.

In this method, the *CoA* of the *MN* is certified using the secure interface ID. It is based on the *MN's* private key and a valid subnet prefix. Please note that this method is only used to determine the *HoA* ownership belongs to the *MN* and nothing else. It does not mean the *MN* also owns the *CoA*. Once *HoA* ownership has been verified only then will the *MN* signs the encrypted *CoA* using the public key of the *CN* with the *HoA* using its private key. Doing this proves the correctness of the *HoA* and *CoA* binding to the *CN*. And the *CN* in turn will verify the *CoA* and *HoA* owner by checking the signature of the *MN*. Then it will compare the *HoA* hash and the hash received from the *HA*. If either one of them receives a negative result, the message will be rejected.

5. Discussion

The method presented here is written in C++ and has been packaged in the Omnet++. The basic functions of the inetmanet-2.0 are imported and the package is configured so that it can be utilised for basic IPv6 functions. The following steps explain the simulated detection:

- The *MN1* can detect if there are no streaming packets coming from the *CN*. It uses the timestamp (Figure 2).
-

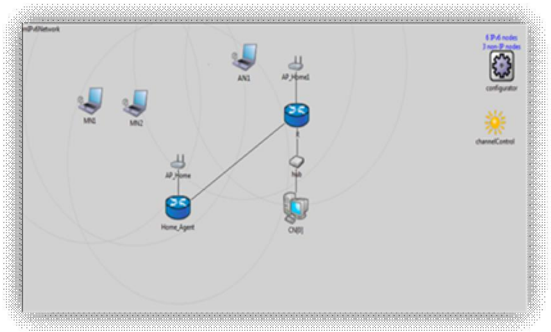

Figure 2. Attacks Simulation for PKBU Protocol in OMNeT ++.

If it suspects a DoS attack, the *BU* can be resent back to the *CN*. The *BC* will be updated with the *HoA* of *MN* and the *CN's* legitimate *CoA*. The packets from the *CN* will now resumed sending to the *MN*.

- The *HoA* of *MN1* can be reclaimed if the *AN* sends a fake *BU* to the *CN*. It can do this by sending a different *CoA* in the *BU* message. The *CN* will proceed to compare the hash value of the *HoA* in the *BU* message with the original hash value of the *HoA*. If the comparison result is negative, that means the *MN* does not own the new *HoA*.
- The point of detection could be the *CN* itself when it is updating the *BC* after the *BU* is received. In PKBU, the *BU* packet is decapsulated after the *CN* receives it. It can proceed with ensuring the signature used to sign the *BU* message is valid. According to the PKBU framework, the *MN* signs the *BU* message. The Elliptic Curve Digital Signature algorithm is used to digitally sign the *BU*. If the verification of the signature is correct, the *CN* can be sure that the packet was sent by the *MN*. The digital signature (r, s) of the *BU* message can be verified using the following steps:

1. Acquire a valid copy of the sender public key *(E, P, n, Q)*.
2. Verify r and s are integers in the interval [1, *n*-1].
3. Compute $w = s-1 \bmod n$ and $h(m)$.
4. Compute $u1 = h(m).w \bmod n$ and $u2 = r.w \bmod n$.
5. Compute $u1P+u2Q = (x0 , y0)$ and $v = x0 \bmod n$.
6. The signature is accepted only if $v=r$.

The Elliptic Curve Discrete Logarithm Problem (ECDLP) algorithm provides strong security by merely using smaller keys according to the research outcome of A. Khaled and M. AL-Kayali. As the key is small, the memory required to store it would also be small. This keeps the size of the data transferred between the nodes to a minimum, thus, leading to shorter transmission time. The applications in the *MNs* require strong security features which are only achievable by using long keys. By using ECC, the *MN* can maintain their cost while still be able to provide strong security by using smaller size keys, which can be compared to RSA, simultaneously.

The processing time is reduced significantly in ECC, especially if binary field $GF(2^k)$ is applied. The *MN* can manage the elliptic curve domain parameters and the private key as well as generate the ECDSA signatures. In addition, the system is only limited to 162-bit size curves.

- The *CN* can function as the detection point itself. Once the verification of the signature is done after it de-encapsulate the packet, it will begin to decrypt the message. The *MN* will use the *CN*'s public key to encrypt the message before sending the packet to the *CN*. According to the false binding update and the MITM attack, the *AN* can eavesdrop on the *BU* packet. The entire message has to be signed using the private key belonging to *MN1*, which the *AN* does not have. If the *AN* decides to use parameters other than the original *MN* parameters as a signature, then the *CN* will not able to positively verify the signature since the *CN* uses the *MN's* parameter as the basis for verification.
- Once the *CN* has finished the decryption process, the *MN's HoA* and *CoA* contained in the plaintext, are retrieved. Another function is called by the *CN* to check the correctness of the *HoA*. This is to ensure that the *HoA* and the *CoA* belong to the correct owner. This allows the *CN* to detect another potential point of attack. It is important that the *CN* is sure the packets belong to the right *MN1*.
- In the PKBU framework, the *CN* will receive packets from the *MN1* via the *HA*. The hash of *MN1's HoA* is contained in the packet. It is used by the *CN* to verify the *HoA* ownership. The *CN* will call a function to calculate the *HoA* hash and then compare it with the *HoA* hash value it receives from the *HA*. If the function returns a TRUE value, then it is verified that the sender of the packet is actually *MN1*. The *CN* will

proceed to save the *BC* entry in regard to the *MN*, otherwise the *CN* will consider it as an attack. The verified packet from the *MN* is now accepted by the *CN* and the traffic between the *CN* and the *MN* can now begin with the *MN1*'s new *CoA*.

- This is a classic example of collaborative defeating the DoS attacks. At the end of the simulation, it is evident that the attack was launched by a malicious entity and the legitimate nodes worked together to detect the DoS, MITM, and session hijacking attacks. Using the PKBU protocol, the attack detection can be done in a collaborative manner.
-

6. Conclusion

In order to protect the binding updates against Man-In-The-Middle, false binding and DoS attacks, several security protocols have been specifically designed and modified such as Cryptographically Generated Address (CGA) (Aura 2005), Early Binding Update (EBU) (Vogt, Bless et al. 2005), Purpose-Built Key (PBK) (Bradner, Mankin et al. 2003), CAM-Child-Proof Authentication (Lowe 1997), Unauthenticated Diffe-Hellman-based binding update (Le and Faccin December 2001.), Enhanced Rout Optimization (Arkko, Haddad et al. 2007), Certificate-based binding update (Deng, Zhou et al. 2002), share key (Datta, Derek et al. 2007), Ticket-based (Koo, Koo et al. 2006) and BAKE/2 protocol (Datta, Derek et al. 2007).

By using our proposed method, the connection between the corresponding nodes and the mobile nodes are ensured. A combination of the ideas below has been used in the PKBU protocol:

- Using an algorithm that is INF-less instead of INF-based. Reason being:
  - o The vulnerability of information being centralised in nature is a big disadvantage.
  - o Directory needs to have constant maintenance to ensure its upkeep.
  - o If a third party directory has been successfully breached, then all the nodes information is available to the attackers.
- Using asymmetric cryptography instead of symmetric.
- Using the private and public keys to create the 128-bit IPv6 addresses.
- The *MN* must ensure the *CN* that:
  - o The *MN* owns the *HoA*
  - o The actual *BU* request belongs the *MN*
- By using the validation process without a PKI, the *MN* can ensure the ownership of the *HoA* and *CoA* to the *CN*.
- The ownership of the *HoA* and *CoA* can be tested to ensure that it belong to the *MN*.
- The *MN* sends a hash of the authentication data through the home address to the corresponding node.
- In order to verify the *MN's* reachability at the claimed *CoAs*, the concurrent *CoA* reachability test is applied.

Acknowledgements

This work was supported in part by the University of Malaya, Kuala Lumpur Malaysia under UMRG Grant (RG080/11ICT).

References

[1] Arkko, J., W. Haddad, et al. (2007). Enhanced route optimization for Mobile IPv6.

[2] Arkko, J., P. Nikander, et al. (2002). "Selection of MIPv6 Security Level Using a Hashed Address." Internet Engineering Task Force (IETF).

[3] Arshad, M. J., A. Farooq, et al. (2012). "A Path Towards IP-V6 Transition Strategies for Scientific Research: An Overview." Life Science Journal 9(1).

[4] Aura, T. (2005). "Cryptographically generated addresses (CGA)." In Proc. 6th Information Security Conference (ISC'03) volume 2851 of LNCS: pages 29-43.

[5] Boneh, D. and M. Franklin (2003). "Identity-based encryption from the Weil pairing." SIAM Journal on Computing 32(3): 586-615.

[6] Boyko, V., P. MacKenzie, et al. (2000). Provably Secure Password-Authenticated Key Exchange Using diffie-hellman. in EUROCRYPT'00: Proceedings of the 19th international conference on Theory and application of cryptographic techniques. Berlin, Heidelberg, Springer-Verlag, 2000: pp. 156{171.

[7] Bradner, S., A. Mankin, et al. (2003). A framework for purpose-built keys (PBK).

[8] Datta, A., A. Derek, et al. (2007). Protocol composition logic (PCL), Elsevier. 172: 311-358.

[9] Deng, R. H., J. Zhou, et al. (2002). Defending against redirect attacks in mobile IP. in CCS '02: Proceedings of the 9th ACM conference on Computer and communications security, (New York, NY, USA), ACM: 59-67.

[10] Fu, J.-M., J.-P. Lin, et al. (2011). "Provably Secure Password-based Three-party Key Exchange Protocol with Computation Efficiency." Life Science Journal 8(4): 394-401.

[11] Georgiades, A. (2011). A security protocol for authentication of binding updates in Mobile IPv6, Middlesex University.

[12] Kempf, J., C. Gentry, et al. (2002). "Securing IPv6 neighbor discovery using address based keys (ABKs)." Internet Engineering Task Force (IETF).

[13] Kohl, J. and C. Neuman (1993). The Kerberos network authentication service (v5), RFC 1510, september.

[14] Koo, J., J. Koo, et al. (2006). A New Authentication Scheme of Binding Update Protocol on Handover in Mobile IPv6 Networks, Springer: 972-978.

[15] Le, F. and S. M. Faccin (December 2001.). Dynamic Diffie Hellman based Key Distribution for Mobile IPv6. Expired Internet-Draft: draft-le-mobileip-dh-01.txt. 4.

[16] Lowe, G. (1997). Casper: A compiler for the analysis of security protocols, IEEE: 18-30.

[17] Menezes, A. J. (1993). Elliptic curve public key cryptosystems, Springer.

[18] Miller, V. (1986). Use of elliptic curves in cryptography. in: Proceeding of the Lecture notes in computer sciences; 218 on Advances in cryptology CRYPTO 85, Springer-Verlag New York, USA.

[19] Perkins, C. E. (1997). Design Principles and practices, Addison-Wesley Longman Publishing Co., Inc., Boston, MA.

[20] Rivest, R. L., A. Shamir, et al. (1978). "A method for obtaining digital signatures and public-key cryptosystems." Communications of the ACM 21(2): 120-126.

[21] Tanenbaum, A. S. and M. Van Steen (2003). "Distributed Systems Principles and Paradigms." Network 3: 26.

[22] Vogt, C., R. Bless, et al. (2005). Early Binding Updates for Mobile IPv6, in Wireless Communications and Networking Conference, 2005 IEEE. 3: 1440-1445 Vol. 1443.

[23] Werapun, W. and A. Unakul (2004). Secure Mobile IPv6 Binding Updates with Identiy-based Signature. international conference on Electronics Packaging.

[24] Xiao, H. (2003). Trust management for mobile IPv6 binding update. Proceedings of the International Conference on Security and Management, Las Vegas, SAM, CSREA Press.

5/22/2013