Modified CGA for Frequently Moving Mobile Nodes in Secure Neighbor Discovery

Esther Kim*, Nam-Uk Kim**, Soo-Duek Kim**, Tae-Myoung Chung*
* Dept. of Computer Science, Sungkyungwan University
**Dept. of Electrical and Computer Engineering, Sungkyungwan University
e-mail : tmejsla86@skku.ac.kr, nukim@imtl.skku.ac.kr, sdkim@imtl.skku.ac.kr, tmchung@ece.skku.ac.kr

Abstract

IPv6 is newly introduced to solve limitations and problems of IPv4 and in IPv6 network, nodes use Neighbor Discovery protocol to discover the subnet prefix and configure its own address. However, Neighbor Discovery is vulnerable to various attacks as it does not have secure mechanism to protect itself. Thus, the Secure Neighbor Discovery has introduced and the main mechanism used in Secure Neighbor Discovery is Cryptographically Generated Address. In this paper, we provide a brief of Cryptographically Generated Address and its limitation in a case where a mobile node moves from one network to another frequently. The proposed scheme resolves this limitation by using the fixed interface identifier.

1. Introduction

IPv6 is the new version of Internet Protocol and designed as the successor of IPv4 to provide larger address space and higher level of security, mobility and other functionalities. In IPv6 network, nodes use Neighbor Discovery(ND) which provides Router Discovery, Prefix Discovery, Address Resolution, Neighbor Unreachability Detection, Duplicate Address Detection(DAD) and Redirect functions and generate their address by prepending the prefix to the interface identifier via Stateless Address Auto-configuration. However, ND is vulnerable to various attacks as messages are not secured. Though the RFCs specify to use IPsec to protect Neighbor Discovery Protocol(NDP) messages, they do not instruct the detailed method. Besides, IPsec can only be used with a manual configuration of security association due to bootstrapping problems. To solve the security issue, Secure Neighbor Discovery[1], which uses Cryptographically Generated Address(CGA)[3], is designed to protect NDP messages. CGA is a technique that creates an address by hashing the address owner's public key via a cryptographic hash function to assert address ownership. However, CGA takes a quite time to generate an address due to complicated computation and this shows that CGA has a limit in a case where a mobile node moves several networks frequently. In this paper, we propose a scheme, which may complement such limitation.

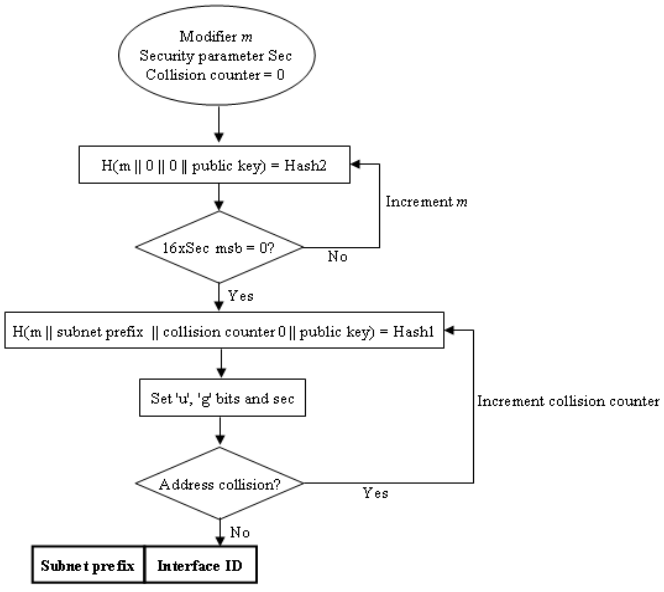
The paper is organized as follows. Section 2 and section 3 describe CGA and the proposed scheme including address generation process respectively. Section 4 explains how the proposed scheme improves CGA in terms of efficiency and security. Finally, section 5 gives a conclusion.

2. Related work

2.1 Cryptographically Generated Address

CGA is IPv6 address of which the interface identifier is generated by computing a cryptographic hash function from the public key and other parameters. Each CGA is associated with CGA parameters data structure: modifier, subnet prefix, collision count, public key and optional extension fields. CGA is formed by prepending the prefix to the interface identifier generated via the hash function. The detailed generation process is shown in the figure 1.

Basically, CGA enables nodes to generate their own address, prove their ownership and verify whose from others without any certification authority or security infrastructure. A node sends the public key and a signed message from the CGA address and a receiver recomputes the hash of the public key, compares the hash with the interface identifier of the source address and verifies the signature using the public key. In this way, the receiver confirms that the message is sent by the owner of the source address and thus, CGA prevents stealing and spoofing of existing IPv6 address, meaning that it prevents impersonation.



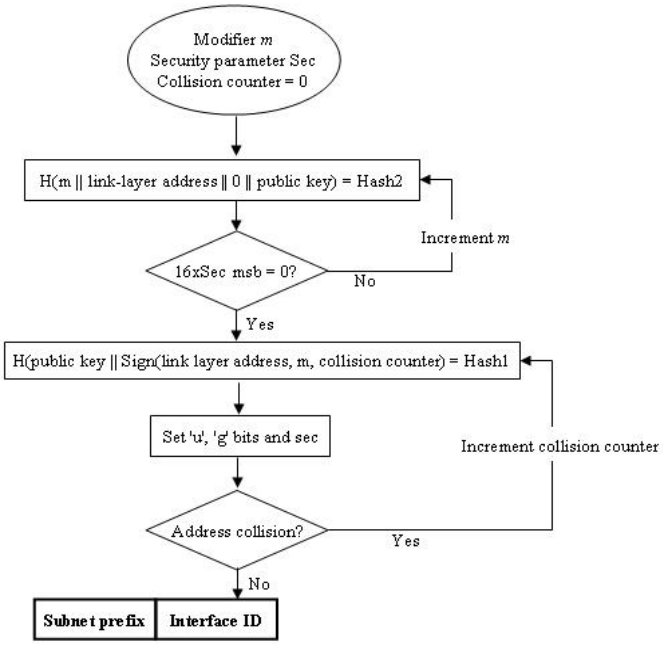
(Figure 1) CGA generation

However, the problem is that though CGA prevents spoofing of someone's address, an attacker can generate a new address with any subnet prefix and CGA does not prove whether the node or address actually exists or not. Besides,

for a mobile node, which moves frequently, needs to generate new address every time the subnet prefix changes.

3. Proposed scheme

The main idea of proposed scheme is to fix the interface identifier by using a link-layer address to reduce the computational delay while enhancing the lack of authentication in the verification process. For that, the link layer address is included in both domains of Hash1 and Hash2 instead of the subnet prefix. The link layer address can be added to the extension field of CGA parameters data structure without any other changes in the data structure. The generation is as shown in the figure 2.



(Figure 2) Modified CGA generation

The address owner signs the link-layer address, modifier and the collision counter with his private key and the public key is concatenated to the signature and the interface identifier is obtained by hashing this concatenation. Then, the receiver starts verification by checking the subnet prefix first, then extract modifier, collision count and the link layer address by verifying the signature. In addition, the address owner can perform the optimistic DAD, which enables the node to use the address while performing DAD.

4. Performance Evaluation

We evaluate the modified CGA in three perspectives.

First of all, for a mobile node, which travels from one network to another frequently needs to recompute the address whenever the subnet prefix changes. Original CGA takes 1.7964 seconds to generate an address on P3-693MHz[7] and this is a quite long time for the node which moves frequently as such frequent movement require the generation every time it moves and this may cause delay in communication or transmission of data, as the node or other node, for which the node is communicating with, has to wait until the address is generated. Plus, if the mobile node is moving by train or any other that moves fast, may not have much time to generate the address. For this reason, this paper proposed to use a fixed interface identifier by using the link layer address rather than the subnet prefix for computation of hash and in this way, delay in computing the hash function with the subnet prefix would be eliminated. Also, the optimistic DAD can be used based on the fact that the possibility of duplicate address is quite low and in this way, the delay in DAD would also be eliminated.

Secondly, by using the link layer address, the binding between the link layer address and the IP address can be created and thus, in the verification process of CGA, the link layer address can be proved that it is bound to the owner of IP address and this would eliminate the lack of authentication of link layer address.

Lastly, by signing the associated parameters and using the signature with the public key in the hash function, the authentication of address can be enhanced as it proves that only the owner of private key make the signature. Therefore, the attacker cannot impersonate though he found the values of parameters and the public key as the binding between the public key and the address is strengthened. Also, in this way, it complements the weakness caused by fixing the interface identifier as no one else except the one with the private key can create an address and this signature proves the certain owner of the address. In addition, though an attacker creates an address to send bulk messages or attack other nodes, he cannot deny that he has sent those messages if he is found to be the attacker. Hence, it would eliminate the lack of authentication of IP address and enhance the proof of address ownership.

5. Conclusion

In this paper, we have presented brief of CGA and its limitation and a proposal to solve such limitations and enhance the security issues. The modified CGA suggests to use link layer address in both domain of Hash1 and Hash2 to make a fixed interface identifier so that a mobile node can reduce unnecessary computation work while it moves around and to create a binding between the IP address and the link layer address for enhancing the authentication. Finally, it suggests to sign the associated parameters and use the signature together with the public key in the Hash1 to enhance the authentication and provide non-repudiation property so that any attacker cannot deny that the address is actually belongs to him.

References

[1] J.Arkko, “Secure Neighbor Discovery”, RFC 3971, March 2005

[2] P.Nikander, “IPv6 Neighbor Discovery(ND) Trust Models and Threats”, RFC 3756, May 2004

[3] T.Aura, “Cryptographically Generated Addresses(CGA)”, RFC 3972, March 2005

[4] M.Bagnulo, “Support for Multiple Hash Algorithm in Cryptographically Generated Addresses(CGAs)”, RFC 4982, July 2007

[6] Joppe W.Bos, “Analysis and Optimization of Cryptographically Generated Addresses”, IEEE, 2008

[7] 박기태, “IPv6 의 보안 기능을 강화하는 Secure ND Protocol 의 구현”, 정보과학회논문지, vol.2, Dec, 2005