Network Working Group H. Rafiee
INTERNET-DRAFT Huawei Technologies Duesseldorf GmbH
Intended Status: Informational Track C. Meinel
Hasso Plattner Institute
Expires: February 11, 2015 August 11, 2014

Possible Attack on Cryptographically Generated Addresses (CGA)
<draft-rafiee-6man-cga-attack-02.txt>

Abstract

This document describes the possible attacks with the use of Cryptographically Generated Addresses.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on February 11, 2015.

Table of Contents

1. Introduction

Cryptographically Generated Addresses (CGA) [RFC3972] is one of the important options of Secure Neighbor Discovery (SeND) [RFC3971] in IPv6 networks. CGA provides the node with the proof of IP address ownership by finding a binding between the public key and the node's IP address. Therefore, It can protect the nodes from network layer IP spoofing attack and prevent forging the identity (if it is only based on the IP address). However, CGA, itself is vulnerable to some types of attacks such as DoS, replay attack (The use of timestamp would mitigate this attack), etc [3]. The goal of this document is not to focus on the well-known attacks but the CGA vulnerabilities that is the result of the text in CGA specification to warn implementers about these possible attacks.

2. Sec value vulnerability

CGA values are the fingerprint of public key. They are generated by executing a hash function on public key and some other parameters. Since the default algorithm for generating this hash is SHA-1, the attacker node only needs to do brute force attacks against 59 bits. CGA algorithm uses sec value (a value between 0 to 7) to increase the brute force search space from 59 bits to maximum 171 bits (59+sec*16) and as a result complicates the brute force attacks to break CGA. Nevertheless, in practice, only sec value 0 and 1 can be used because it takes hours to years to generate CGA sec value higher than 1 [2].

Unfortunately, in practice, it does not matter what sec value the victim node chooses and the use of sec value only complicates the IP address generation process for the victim node. This is because the attacker will only use sec value 0 and SHA1 algorithm.

It is true that the attacker has a source IP address that is only in 3 bits differs from the victim node's source IP address. But since the node only checks the target address and start the verification process on the source address, this really doesn?t bother the attacker and the attack is successful. The attacker node can persist on his own IP address after a successful verification by CGA node and either force CGA node to generate a new IP address and again the attacker repeats this process or there will be duplicate addresses on the network which cause many services in the victim network stop working. This is because all the nodes verify this attacker node the same way as the legitimate CGA node processed the verification. From their aspects, these two nodes are the same.

The mentioned flaw occurs during verification processes in all verifier nodes. The node needs to verify other nodes in two different

conditions -- during DAD process and during checking the neighbors' reachability in cache. This means that the CGA security is only the security of CGA sec value 0 that is 2^59 bits.

The reason are as follow:

1- Section 5 RFC 3972

It uses the term ?the address?, The address refers to the source address. (please refer to Number 4 to know the reason)

2- step 4, Section 5 RFC 3972 :The CGA verifier node ignores 3 bits sec value in source address and 2 bits u and g

Based on NDP specification, the verifier node checks to see whether or not the target address is the same as its own IP address. If it is the same and the node supports CGA, then it starts CGA verification. Based on step 4 section 5 RFC 3972, the CGA node compares the source address (IID section) of the sender node to his own IID. The verifier node ignores 3 bits sec value. So, the attacker can set the target address to the real CGA address of the victim node disregard its sec value and set the source address to his own CGA value that is only different in the 3 leftmost bits. Since the verification is successful, the attacker can spoof the IP address of CGA node.

3 - No comparison of source address and target address

Based on the Neighbor Discovery Protocol (NDP) specification on section 7 RFC 4861 [RFC4861, RFC4862], there is nothing about to compare the source IP address with the target address. In SeND specification [RFC3971], there are rules for the sender node. However, the verifier node never checks those rules. This is why the attacker can ignore them. So, the attacker can create the SeND message by using his own CGA address that differs only in sec value. The attacker selects the victim node's source address as his own target address and sends this message.

4- Section 5.1.2 RFC 3971

"If the interface has been configured to use CGA, the receiving node MUST verify the source address of the packet by using the algorithm described in Section 5 of [11]. The inputs to the algorithm are the claimed address, as defined in the previous section, and the CGA Parameters field.".

This is where I explained in Number 1 that "the address" refers to the source IP address.

5- Section 7.2.3 RFC 4861

SeND only adds 4 options to NDP but does not change the initial way to process the NDP messages.

"- The Target Address is a ?valid? unicast or anycast address

assigned to the receiving interface [ADDRCONF],

- The Target Address is a unicast or anycast address for which the

node is offering proxy service, or

- The Target Address is a ?tentative? address on which Duplicate

Address Detection is being performed [ADDRCONF]."

It is usually the new CGA victim node that needs to process Duplicate Address Detection (DAD). In other words, the attacker has a valid unicast target address that is similar to what is chosen by the victim CGA node. There is no place in the standard NDP specifications to explain that the target address should be compared to the source address. The reason is because sometimes the target address is temporary. Unfortunately in SeND specification, this check was not done too because it only follows ND specification.

6- Section 7.2 RFC 3972: SeND cannot also protect the node against this problem

The author of CGA specification was aware of this problem:

"For each valid CGA Parameters data structure, there are 4*(Sec+1) different CGAs that match the value. This is because decrementing the Sec value in the three leftmost bits of the interface identifier does not invalidate the address, and the verifier ignores the values of the ?u? and ?g? bits. In SEND, this does not have any security or implementation implications."

But the assumption was that SeND can protect the nodes from this attack by the use of a valid certification. But unfortunately, certification is only to authorize routers and not for all nodes in the network. The other solution is to use Microsoft Active Directory (AD). But in practice not all places use or will use this approach.

3. Key size Vulnerability

The lower limit for key size is 384 bits. The attacker can choose the lowest size public key so that he can better play with the public key bits and easier and faster generates the similar hash of the CGA node.

4. Modifier can be zero

The attacker does not need to generate a really good random value. Since for him it is only important to match the hash value. This is

especially true for the scenario where the attacker needs to do brute force attacks against all 64 bits and sec value is not ignored.

5. Variable length Prefixes

The assumption in CGA algorithm is that the subnet prefix is 64 bits. This really makes the verification process easier and straight forward. But if any networks wants to have a variable length prefix, then CGA verifier node must know which part of this address is IID and which part is prefix. If it can receive this information from an authorized router, then there might be no risk for the verifier node. But if this value supposed to receive from the sender node, then the problem would be where to add such information. If it is as a new option in CGA, then the attacker can spoof this value and sign it with its own private key and claim different prefix. But if this value is as a part of IID, then the problem would be the number of bits required to carry the prefix length. This process will decrease the number of bits to carry the CGA value and will lead to reducing CGA security. (59- number of bits to carry the prefix length)

6. Use case Scenario for CGA attack

In the following subsections, some of these attacks are explained in more detail.

6.1. Duplicate Address Detection Process

When a node generates his IP address, it process the DAD in order to avoid collision on the network. The attacker might be able to generate the CGA value the same of the legitimate CGA node and claim the ownership of that IP address. The CGA nodes only tries 3 times and then it gives up.

6.2. Nodes communications

When two nodes want to start communication, they try to find the IP address of eachother by sending multicast NS/NA messages. The attacker can offline generates the value of victim which differs only in 3 bits sec value and then impersonate the victim node and try to communicate with other node. This attack is likely possible especially when the attacker can send this response faster than the real node or the real node is offline at the time of this request by other node.

7. Security Considerations

-

8. IANA Considerations

-

9. Appendix

- CGA multicore attack

This is where you can find CGA attacks (multicore).

http://www.hpi.uni-potsdam.de/meinel/security_tech/ipv6_security/ipv6ssl.html

10. Acknowledgements

The author would like to acknowledge Fabian Braeunlein, one of a bachelor student at Hasso Plattner Institute who assists us, during this busy moments, for writing the attacking codes.

11. References

11.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC3972] Aura, T., "Cryptographically Generated Addresses (CGA)," RFC 3972, March 2005.

[RFC3971] Arkko, J., Kempf, J., Zill, B., and P. Nikander, "SEcure Neighbor Discovery (SEND)", RFC 3971, March 2005.

[RFC4861] Narten, T., Nordmark, E., Simpson, W., Soliman, H., "Neighbor Discovery for IP version 6 (IPv6)", RFC 4861, September 2007.

[RFC4862] Thomson, S., Narten, T., Jinmei, T., "IPv6 Stateless Address Autoconfiguration", RFC 4862, September 2007.

[1] AlSa'deh, A., Rafiee, H., Meinel, C., "Cryptographically Generated Addresses (CGAs): Possible Attacks and Proposed Mitigation Approaches," in proceedings of 12th IEEE International Conference on Computer and Information Technology (IEEE CIT'12), pp.332-339, 2012.

[2] Bos, J., Oezen, O., Hubaux, J., "Analysis and Optimization of

Cryptographically Generated Addresses", In Proceedings of the
12th International Conference on Information Security (2009),
ACM, pp. 17 ? 32.

[ugbits] Carpenter, B., Jiang, S., "Significance of IPv6 Interface Identifiers", RFC 7136, February 2014.

[variableprefix] Carpenter, B., Chown, T, Gont, F., Jiang, S., Petrescu, A., Yourtchenko, A.," Analysis of the 64-bit Boundary in IPv6 Addressing", http://tools.ietf.org/html/draft-ietf-6man-why64 , April 2014

Authors' Addresses

Hosnieh Rafiee
HUAWEI TECHNOLOGIES Duesseldorf GmbH
Riesstrasse 25, 80992,
Munich, Germany
Phone: +49 (0)162 204 74 58
Email: hosnieh.rafiee@huawei.com

Christoph Meinel
Hasso-Plattner-Institute
Prof.-Dr.-Helmert-Str. 2-3
Potsdam, Germany
Email: meinel@hpi.uni-potsdam.de