Network Working Group
Internet Draft
Intended status: Informational
Expires: September 16, 2012

Sheng Jiang
Huawei Technologies Co., Ltd
Sean Shen
CNNIC

March 12, 2012

Analysis of Possible DHCPv6 and CGA Interactions

draft-ietf-csi-dhcpv6-cga-ps-09.txt

Status of this Memo

Copyright Notice

Abstract

This document analyzes the possible interactions between DHCPv6 and Cryptographically Generated Addresses (CGAs), and gives recommendations on whether or not these interactions should be developed as solutions.

Table of Contents

1. Introduction

The Dynamic Host Configuration Protocol for IPv6 (DHCPv6) [RFC3315] can assign addresses statefully. Although there are other ways to assign IPv6 addresses [RFC4862, RFC5739], DHCPv6 is also used when network administrators require more control over address assignments or management to hosts. DHCPv6 can also be used to distribute other network configuration information from network to hosts.

Cryptographically Generated Addresses (CGAs) [RFC3972] are IPv6 addresses for which the interface identifiers are generated by computing a cryptographic one-way hash function from a public key and auxiliary parameters. Associated with public & private key pairs, CGAs are used in protocols, such as SEND [RFC3971] or SHIM6 [RFC5533], to provide address validation and integrity protection in message exchanging.

As an informational document, this document analyzes the possible interactions between DHCPv6 and Cryptographically Generated Addresses (CGAs), and gives recommendations and reasons whether these possibilities should be developed as solutions or be declined in the future. This document itself does NOT define any concrete solutions.

Firstly, the scenario of using CGAs in DHCPv6 environments is discussed. Then, configuring CGA-relevant parameters using DHCPv6 is discussed. Although CGA generation delegation is considered not suitable for DHCPv6, it is also analyzed. Security considerations for proposed interactions are examined.

2. Coexistence of DHCPv6 and CGA

CGAs were designed for SeND [RFC3971]. The CGA-associated public key, which is also transported to the receiver, provides message origin validation and integrity protection without the need for negotiation and transportation of key materials. SeND is generally not used in the same environment as a DHCP server.

However, after CGA has been defined, as an independent security property, many other CGA usages have been proposed and defined, such as SHIM6 [RFC5533], Enhanced Route Optimization for Mobile IPv6 [RFC4866], etc. In these scenarios, CGAs may be used in DHCPv6-managed networks.

A CGA address is generated by a host that owns the associated key pair. However, hosts in DHCPv6-managed network get their addresses

from DHCPv6 servers. For a DHCPv6-managed network, CGA owners could be declined network access.

Although the current DHCPv6 specification [RFC3315] has a mechanism that allow a host to request the assignment of a self-generated address from DHCPv6 servers, "DHCPv6 says nothing about details of temporary addresses like lifetimes, how clients use temporary addresses, rules for generating successive temporary addresses, etc." (quoted from Section 12 [RFC3315]. There is no existing operation to allow DHCPv6 servers to decline the host-requested address and to reply with information to generate a new address.

New DHCPv6 options could be defined to allow DHCPv6 servers to decline requested-CGAs, to inform the host about why the address has been declined, and to give information needed to construct an acceptable CGA.

Specifically, a node could request that a DHCPv6 server grants the use of a self-generated CGA by sending a DHCPv6 Request message. This DHCPv6 Request message contains an IA option including the CGA address. Depending on whether the CGA satisfies the CGA-related configuration parameters of the network, the DHCPv6 server can then send an acknowledgement to the node to either grant the use of the CGA or to indicate that the node must generate a new CGA with a suggested CGA-related configuration parameters of the network. In the meantime the DHCPv6 server may log the requested address/host combination, which completes CGA registration operation.

3. Configuring CGA-relevant parameters using DHCPv6

In the current CGA specifications, it is not possible for network management to influence the CGA generation. Administrators may want to be able to configure or suggest parameters used to generate CGAs. For example, if a network only accepts the network access requests from hosts that use CGAs with Sec value 1 or higher for security reasons, this information should be able to propagate to hosts.

The CGA associated Parameters used to generate a CGA includes several parameters [RFC3972]:

- a Public Key. The key pair is generated by CGA owner. For security reasons, the key pair, more specifically the private key, should not be transported through networks. Centrally managed/generated key is conflict with primary CGA concept. Therefore, a mechanism to configure/suggest a value is not analyzed.

- a Prefix. The prefix can be obtained though Router Advertisement messages of neighbor discovery protocol. DHCPv6 may provide another mechanism to propagate the prefix information to the host. This may enable the CGA usage scenarios without ND attendance.

- a 3-bit security parameter Sec. It is possible that networks request hosts to use CGAs with high Sec value for secure access. However, it is dangerous to allow network to enforce hosts to generate new CGAs with high Sec value, particularly the generation with high Sec value is extremely computational consumption, as analyzed in Section 5. A reasonable compromise could be the network gives suggested information for Sec value, only when the access requests from host are declined for low Sec value.

- a Modifier. This is generated during the CGA generation procedure. Therefore, a mechanism to configure/suggest a value is not analyzed.

- a Collision Count value. This is generated during the CGA generation procedure. Therefore, a mechanism to configure/suggest a value is not analyzed.

- any Extension Fields that could be used. So far, there is no concrete use case for this parameter. If new Extension Fields are defined in the future, whether they are suitable to network-managed configuration should be carefully analyzed based on the specific case.

4. Using CGA to Protect DHCPv6

DHCPv6 is vulnerable to various attacks, e.g. fake address attacks where a "rogue" DHCPv6 server responds with incorrect address information. A malicious rogue DHCPv6 server can also provide incorrect configuration to the client in order to divert the client to communicate with malicious services, like DNS or NTP. It may also mount a Denial of Service attack through mis-configuration of the client that causes all network communication from the client to fail. A rogue DHCPv6 server may also collect some critical information from the client. Attackers may be able to gain unauthorized access to some resources, such as network access. See Section 23 [RFC3315].

In the basic DHCPv6 specifications, regular IPv6 addresses are used. However, DHCPv6 servers, relay agents and clients could use host-based CGAs as their own addresses. A DHCPv6 message (from either a server, a relay agent or a client) with a CGA as source address can

carry the CGA Parameters data structure and a digital signature. The receiver can verify both the CGA and signature, then process the payload of the DHCPv6 message only if the validation is successful. A CGA option with an address ownership proof mechanism and a signature option with a corresponding verification mechanism would allow the receiver of a DHCPv6 message can verify the sender address of the DHCPv6 message, which improves communication security of DHCPv6 messages. CGAs can be used for all DHCPv6 messages/processes as long as CGAs are available on the sender side.

Using CGAs in DHCPv6 protocol can efficiently improve the security of DHCPv6. The address ownership of a DHCPv6 message sender (which can be a DHCPv6 server, a reply agent or a client) can be verified by a receiver. Also, the integrity of the sent data is provided if they are signed with the private key associated to the public key used to generate the CGA. It improves the communication security of DHCPv6 interactions. The usage of CGAs combining with signature verification can also avoid DHCPv6's dependence on IPsec [RFC3315] in relay scenarios. This mechanism is applicable in environments where physical security on the link is not assured (such as over certain wireless infrastructures) or where available security mechanisms are not sufficient, and attacks on DHCPv6 are a concern.

The usage of CGAs can prove the source address ownership and provide data integrity protection. Furthermore, CGAs of DHCPv6 servers may be pre-notified to hosts. Then, hosts can decline the DHCPv6 messages from other servers. But in this case the address will be fixed. It may increase the vulnerability to, e.g., brute force attacks against. The pre-notification operation also needs to be protected, which is out of scope.

5. Computation Delegation of CGA generation

As analyzed in this section, CGA generation may be computationally intensive when Sec value is set to be high. However, DHCPv6 servers are normally not computationally powerful enough to also generate CGAs for all of its hosts.. Particularly, a DHCPv6 server is architecturally designed to serve thousands hosts simultaneously.

In the CGA generation procedure, the generation of the Modifier field of a CGA address is computationally intensive. This operation can lead to apparent slow performance and/or battery consumption problems for end hosts with limited computing ability and/or restricted battery power (e.g. mobile devices). As defined in [RFC3972], the modifier is a 128 unsigned integer that is selected so that the 16*SEC leftmost bits of the second hash value, Hash2, are zero. The modifier is used during CGA generation to implement

the hash extension and to enhance privacy by adding randomness to the CGA. The higher the number of bits required being 0, the more secure a CGA is against brute-force attacks. However, high number of bits also results in additional computational cost for the generation process, cost that could be deemed excessive. As an example, consider a Sec value equals 2, requesting the leftmost 32 bits of a SHA-1 Hash2 to be zero. For assuring this, a system has to generate in mean 2^32 different modifiers, and perform the Hash2 operation to check the bits required to be 0. An estimation of the CPU power required to do this can be obtained as following: openSSL can perform in an Intel Core2-6300 on an Asus p5b-w motherboard close to 0.87 million of SHA-1 operations on 16 byte blocks per second. Since the input data of Hash2 operation is larger than 16 bytes, this value is an upper bound for the number of hash operations that can be performed for generating the modifier. Checking 2^32 different modifiers requires around 5000 seconds. A practice experimental on a platform with an Intel Duo2 (2.53GHz) workstation showed the results of average CGA generating time as below: when SEC=0, it took 100us; SEC=1, 60ms; SEC=2, 2000s (varies from 100~7000sec). The experiment was unable to be performed for SEC=3 or higher SEC values. Theoretically estimating, about 30000 hours are required to generate a SEC=3 CGA.

Generating a key pair, which will be used to generate a CGA, also requires a notable computation, though this may only be issues on a very low-power host occasionally.

A very low-power host might want to delegate its key and hash generation to a more general purpose computer. In such cases, a mechanism to delegate the computation of the modifier would be desirable. This would be especially useful for large SEC values.

However, DHCPv6 servers are not suitable to serve such computational delegation requests from thousands clients. Correspondently, the security analysis of CGA generation delegation and key generation delegation are out of scope.

6. Conclusion

This document analyzed the possible interactions between CGA and DHCPv6. The analysis has determined that a few interactions are not worth pursuing including: enforcing CGA Sec value, using DHCPv6 to manage CGAs, using DHCPv6 to assign certificates or centrally generated key pair, using DHCPv6 for delegating CGA generation or key generation, etc.

This document suggests a few possible interactions be investigated:

- allowing DHCPv6 servers to decline requested-CGAs and reply with Prefix or Sec values to generate an appropriate CGAs,

- using CGA addresses for interactions between DHCPv6 servers/relays and clients

7. Security Considerations

Allowing DHCPv6 servers to decline the requested-CGA and reply with information to generate an appropriate CGA might actually increase network access flexibility. This might also benefit the network security too.

Prefix is information that can be advertised. However, if DHCPv6 propagates the prefix to hosts, then attackers have another way to propagate bogus prefixes. This can waste hosts' resources. DHCPv6 snooping, DHCPv6 authentication and DHCPv6 server using CGAs can help to prevent or discover bogus prefixes.

When propagating the Sec value from the DHCPv6 server to host, it is only returned if the DHCPv6 declines the requested-CGA. For security reasons, networks should not enforce any CGA parameters. Enforcing CGA parameters could allow malicious attackers to attack hosts by forcing them to perform computationally intensive operations. Networks can suggest the Sec value, but hosts need not heed the suggestion. However, if the hosts do not follow the suggestion, then the network might deny network services, including access services.

Using CGA as source addresses of DHCPv6 servers, relays or, also in DHCPv6 message exchanging provides the source address ownership verification and data integrity protection.

IF a DHCPv6 server rejected a client CGA based on a certain Sec value, it should not suggest a new Sec value either equal or lower than that rejected Sec value.

Without other pre-configured security mechanism, like pre-notified DHCPv6 server address, using host-based CGA by DHCPv6 servers could not prevent attacks claiming to be a DHCPv6 server. Alternatively, IPsec may be used, but it is a heavier security mechanism.

8. IANA Considerations

There are no IANA considerations in this document.

9. Acknowledgements

Useful comments were made by Marcelo Bagnulo, Alberto Garcia, Ted Lemon, Stephen Hanna, Russ Housley, Sean Turner, Tim Polk, David Harrington, Jari Arkko, Tim Chown, Pete Resnick and other members of the IETF CSI working group.

10. Diff from last IESG review (2010-10) [RFC Editor please remove]

Added CGA-DHCP co-existing scenarios, as the second paragraph in Section 2.

Added the need for a DHCPv6 address registration operation in Section 2.

Rewrite the CGA parameters configuration section. Analyze the requirements per CGA parameters.

Added statement that network should not enforce the CGA parameters, but may suggest.

Removed misleading words that linked CGA with PKI.

Removed misleading words central managed CGA.

Removed the combination of CGA and an external authentication/authorization, since it is conflict with primary CGA concept.

Removed the possible operations that DHCPv6 server may assign certificates or centrally generated key pair.

Added statement that CGA generation delegation is not suitable for DHCPv6 servers.

Added a conclusion section so that the message from this document is clearly summarized.

Rewrite the security consideration section. Only focus on proposed operations.

11. References

11.1. Normative References

[RFC3315] R. Droms, Ed., J. Bound, B. Volz, T. Lemon, C. Perkins and M. Carney, "Dynamic Host Configure Protocol for IPv6", RFC 3315, July 2003.

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

[RFC3972] T. Aura, "Cryptographically Generated Address", RFC 3972, March 2005.

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

[RFC4866] J. Arkko, C. Vogt and W. Haddad, "Enhanced Route Optimization for Mobile IPv6", RFC 4866, May 2007.

[RFC5533] M. Bagnulo and E. Nordmark, "Shim6: Level 3 Multihoming Shim Protocol for IPv6", RFC 5533, June 2009.

[RFC5739] P. Eronen, J. Laganier, C. Madson, "IPv6 Configuration in Internet Key Exchange Protocol Version 2 (IKEv2)", RFC 5739, February 2010.

Author's Addresses

Sheng Jiang
Huawei Technologies Co., Ltd
Q14, Huawei Campus
No.156 Beiqing Road
Hai-Dian District, Beijing 100095
P.R. China
Phone: 86-10-82882681
Email: jiangsheng@huawei.com

Sean Shen
CNNIC
4, South 4th Street, Zhongguancun
Beijing 100190
P.R. China
Email: shenshuo@cnnic.cn