A Pull Model IPv6 Duplicate Address Detection

Guang Yao, Jun Bi, Sen Wang
Network Research Center
Tsinghua University
Beijing, China
{yaog,junbi,wangs}@netarchlab.tsinghua.edu.cn

Yueran Zhang
High School of Tsinghua University
Beijing, China
zhangyueran@sina.com

Yitian Li
Beijing No.20 Middle School
Beijing, China
liyitian2009@sohu.com

Abstract—**In IPv6 network, before configuring any address, a node must perform Duplicate Address Detection (DAD) to ensure the address is unique on link. However, original DAD is unreliable and vulnerable. In this article, a pull model DAD is designed, which achieves improvements both in reliability and security through changing the solicitation model. Comparing with SEcure Neighbor Discovery (SEND), this proposal has advantage in lightweight overhead and flexibility of address generation. Through evaluation, it is found to be feasible and cost effective.**

Keywords-IPv6; duplicate address detection; security

I. Introduction

Duplicate Address Detection (DAD) [1] is used to check the uniqueness of a self generated address when IPv6 Stateless Address Autoconfiguration (SLAAC) [1] is enabled, which gives the host the ability to configure interface automatically. DAD is also performed on address assigned through other mechanisms, including DHCP and manual configuration. In DAD procedure, if no response is received after sending one or more solicitation message, a node can configure the target address. Address configuration through DAD is relatively fast and easy. Moreover, the decentralization feature of DAD satisfies well the requirements of wireless ad hoc network. Currently, DAD is universally enabled on host with IPv6 protocol stack.

Original DAD makes assumption that all the nodes on link are trustworthy. However, this condition is not easy given in public accessible networks. If DAD Neighbor Solicitation (NS) is replied by a malicious node continuously, nodes in the network will have to abandon generated addresses and are prevented from address configuration. In addition, because the NS may get lost on the link and the response message may also get lost, a node may configure an address which is used by another node.

There have been a number of solutions to improve the security of DAD process. Secure Neighbor Discovery (SEND) has been proposed and standardized [2], which improves the security of Neighbor Discovery Protocol (NDP) [3], thus the security of DAD is also enhanced. SEND enforces Cryptographically Generated Address (CGA) [4], which makes the NDP messages are self- certificated. The defect of SEND is that CGA generation is uncontrollable and of heavy cost. Moreover, in scenarios that node wants to use a human memorable address, CGA is not unsuitable. Similar with name properties discuss in Zooko triangle model [5], a CGA address cannot achieve human-meaningful with security and global uniqueness simultaneously. A snooping based solution, which sets up bindings between address and lower layer property of nodes and checks address validity based on the bindings, was proposed in [6]. This solution allows human memorable address, however, it doesn't suit ad hoc network because a validating device is needed.

In this article, a pull model DAD is proposed, which mainly focuses on improving the security of DAD model. In this solution, when performing duplicate detection, instead of sending a NS targeting at a specific address onto the link and waiting the pushed message from other nodes, a solicitation is sent to solicit all the assigned addresses with same hash value from concerning nodes. The tentative node then decides from the responses whether the generated address is unique. Because the hash computation is hard to inverse to find all the possible addresses, the tentative address has little probability to be attacked. This mechanism has no requirement on address generation and network structure, thus it has better universality relatively. This mechanism can also be used on address resolution and unreachability detection.

This article first analyzes the threat model of original DAD. Then the detailed pull model solution is described, together with related analyzing work. Evaluation based on simulation is presented in the following section. The last section compares this solution with existing solutions, and concludes the article.

II. Related Work

Pull model mechanism, in which receiver controls the delivery of traffic, has been used to control unwanted traffic, e.g., SPAM [7]. Till now, this principle has not been used on DAD process. This solution is partly similar with bit commitment scheme [8] in model, which allows one to commit a value without exposing it first. SEND [2] still follows the original push model of DAD. SEND enforces generated address must be CGA, which constrains the usage scope greatly. Manually configured address and DHCP assigned address will also need performing DAD as required in RFC4862 [1]. IETF CSI workgroup partly works on conjunction of SEND and DHCP; however, still CGA is enforced in DHCP procedure. SEND can improve the security of Router Discovery process, which is another serious threat in IPv6 network, but not solved in this mechanism. IETF SAVI workgroup works on another path. Address is validated on a

978-1-4244-8389-1/10/$26.00 ©2010 IEEE

snooping device but not by the node on the link. All the SAVI solutions are still in progress.

III. Threat Model of DAD

In original DAD procedure, when a node wants to check the uniqueness of a generated address, it will send one or more NS (by default, one) onto the link. If there is no Neighbor Advertisement targeting at the tentative address received after pushing the NS, it believes the address is not being used, and then it can configure the address; if a NA is received, it believes the address is being used by another node, thus it will abandon the address and generate another address.

The threat model of DAD includes:

- No response for NS != Address is not being used. (Denoted by ***V1***)

Some events may result in no response even when the address is being used, including: a) NS is lost; b) NA is lost; c) The target node is inactive.

Considering the huge address space of IPv6, collision probability of randomly generated address is trivial. However, for manually configured address, which is always easy to memorize, the collision may have a non-trivial probability.

- A response for NS != Address is being used.(***V2***)

Because every node is able to reply the NS, a malicious node can reply with a NA and prevent the tentative node using the address. The tentative node will always trust the NA, and abandon the address. If the DAD process of the node is attacked continuously, it will fail in getting any address. A malicious node has the ability to stop any other nodes on link from configuring any address.

The threat ***V1*** and ***V2*** of DAD is thought to be resulted from using push model, which is of no acknowledgement mechanism and exposes the target address to malicious node directly.

IV. Mechanism Design and Analysis

A. Overview

As described in Figure 1., in this solution, whenever the host generates an address, or configured manually or from DHCP, instead of sending a DAD NS to check the uniqueness of the address, it computes a hash value of the address, and ask whose address has the same hash value on the link. After knowing all the possible colliding addresses, it checks whether the tentative address is in the list. If no collision, it can configure the address; else it may generate another address and repeat this process.

B. Threat Model Analysis

Because this pull model gives the tentative node the ability to hide its interested address in detection, the tentative address cannot be attacked directly. In order to find the tentative address from detection packet, a malicious node will have to inverse the hash computation, and find all the addresses having the same hash value. If only the hash computation is one-way, the inverse computation will cost heavily. And because of the huge space of IPv6 address, even the addresses having the same hash will still have a huge number; then it will be very hard for the malicious node to attack this process even though flooding all possible addresses. Thus, ***V2*** can be handled.

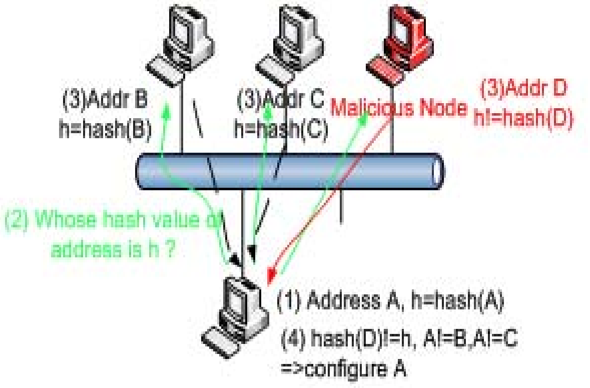

Figure 1. Procedure of Pull Mode DAD

The ***V1*** problem cannot be handled through applying pull model directly. To mitigate this problem, at least one response is expected after one solicitation. This response message can be regarded as an acknowledgement of the solicitation. Then the tentative node knows the solicitation is sent successfully. To achieve this goal, there are some requirements on the hash computation discussed in the following section.

C. Hash Computation

To perform this procedure, a hash function must be deployed on all nodes. This hash function takes generated address as input, and returns a fixed length bit string.

A perfect one way hash can prevent a malicious node from inverting the computation; however, considering the input and output string of this hash function are too short to prevent a brute force attack to find a collision, a practical requirement is that the hash function should be able to prevent an attacker from exhausting all the addresses with a target hash value. The choice of hash function remains our future work.

The output bit string length is a sensitive problem for this mechanism. If the hash value is too long, the collision space will be small and this mechanism will be very vulnerable to any possible inverting attack. If the hash value is too short, nodes with collision address will have a high proportion. Then a solicitation will have to expect a number of responses, the delay of configuration will be increased. Moreover, attacker may make use of such protocol to disturb the network nodes or launch some kind of reflection attack.

An additional requirement is that a tentative node should expect a certain probability to get at least one response. The response can be used as a confirmation that the solicitation is sent successfully. To assure at least one response will be received after each solicitation, some assumption on address generation must be made and the hash function must be designed to satisfy that each hash group has at least one existing address. The hash function design is left as our future work.

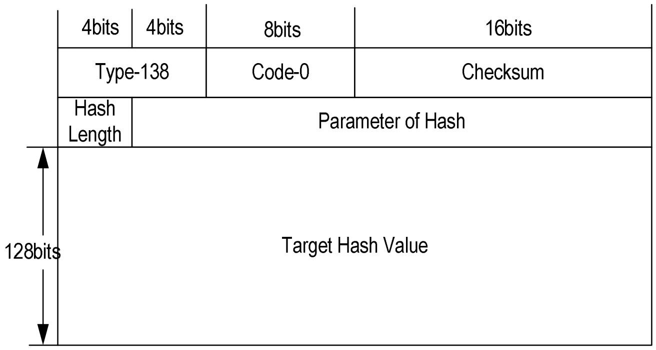

Figure 2. Format of hash solicitation message

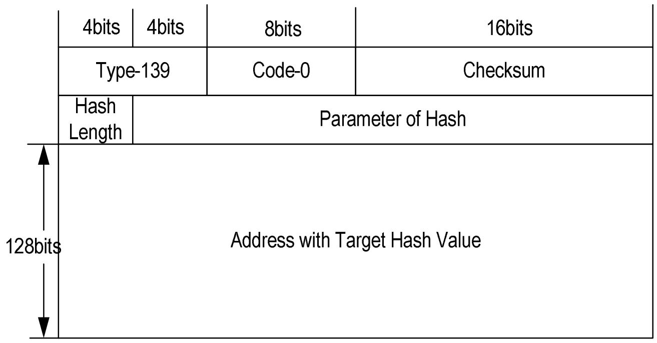

Figure 3. Format of address advertisement message

D. *End of DAD Procedure*

Because a malicious node may perform inverting computation and advertise collision addresses continuously, a tentative node must end receiving response after a period to avoid endless receiving. A node should stop receiving response after a specified time interval which is related to network condition. The interval should also be affected by the output length of hash computation. Through setting this interval, this mechanism can be less sensitive to inverting computation of the hash function.

E. *Multicast Group and Message Type*

Although the Solicitation message can be sent to all-node-multicast group, a large portion of nodes will be disturbed unnecessarily. A new type of multicast group is designed in this proposal. This type of group is identified by the hash value of generated address, instead of the last bits. Whenever a node configures an address, it must join the multicast group identified by the hash value of the address and receive solicitation sent to this group. And it must quit the group if abandoning the address.

Additional message types are also design to cover the semantic change both in solicitation and advertisement. The format of hash solicitation message is described in figure 2. The Hash Length field contains the length of the actual hash value in Target Hash Value field. The Parameter of Hash field is used to contain the possible parameter associated with the hash computation, e.g., a nonce. In current design, no parameter is required, and the input of hash computation is only the address.

For the response message of solicitation (address advertisement message) described in figure 3., although more than one address can have the same hash value on one node, a hash advertisement message is designed to only contain one of them. If a node has multiple addresses with the target hash value, it must send address advertisements for each address.

F. *Address Resolution and Unreachability Detection*

The pull model can also be applied on address resolution and unreachability detection. Instead of sending a solicitation exposing the target address, a hash value is used in solicitation. The soliciting node can find the actual target address from the response messages.

However, because the active addresses on a link are limited, a malicious node can compute all the hash values previously, and map the hash value to existing addresses. Once a hash solicitation is received, it can find the target address immediately. Thus, this mechanism cannot improve the security of address resolution and unreachability detection significantly.

V. EVALUATION

A. *Evaludation Objective and Analysis Parameter*

We first analyzed whether this solution is feasible to meet the requirements in design. Then we compared this mechanism with existing mechanisms on performance.

Suppose N nodes on the link. Each node has M addresses. The output length of hash function is l . We make an assumption that the result of hash computation is uniformly distributed.

B. *Feasibility: Output Length of Hash Computation*

The size of collision space of each address:

$$S = 2^{128-l} \quad (1)$$

The probability of receiving at least one response:

$$P = 1-(1-2^{-l})^{MN} \quad (2)$$

The proportion of disturbed nodes:

$$D = 1-(1-2^{-l})^{M} \quad (3)$$

To ensure a certain probability that at least one response is received after a solicitation, we require $P \geq 0.75$. Then we have $l \leq -\log_2(1-0.25^{1/MN})$. The relationship between upper bound of l, N, M is plotted in Figure 4. It can be found that l has a reasonable upper bound. Relationship between disturbed node proportion, hash length l and M is plotted in Figure 5. This figure shows the disturbed proportion can be controlled through adjusting the value of l. Because l has a typical value between 2 and 16, the size of collision space is still large to prevent inverting attack.

Because the reasonable value of l is determined by the size of network, it is suggested the length of l should be advertised in Router Advertisement.

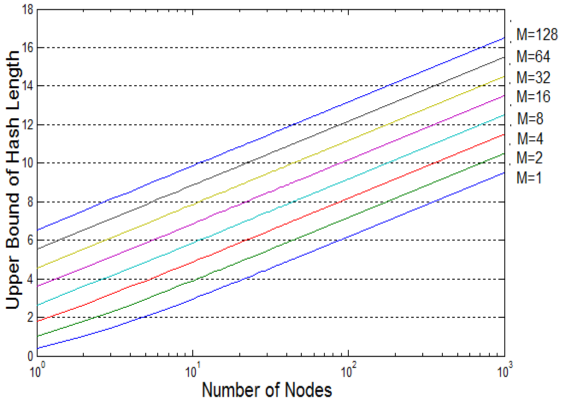

Figure 4. Relationship between upper bound of hash length and network size.

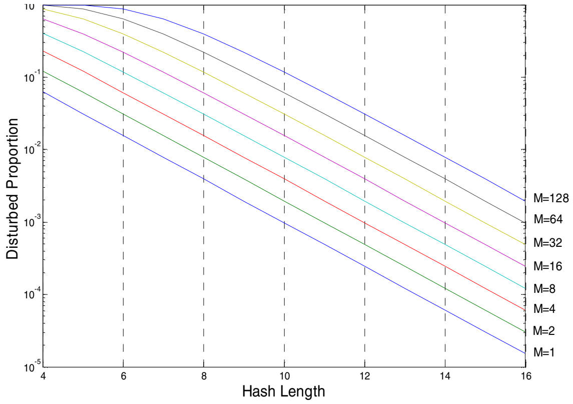

Figure 5. Relationship between hash length and disturbed proportion.

C. *Security Cost Ratio*

To compare this mechanism with existing solutions, we define an index named Security Cost Ratio (SCR). It is computed through dividing the attack cost by address generation cost.

For simplification, we assume a malicious node would have search in a space of 2^{128-l} to block a node enabled this mechanism from configuration. Actually, if the hash computation is one way function, the malicious node will have to search in a space of 2^{128}. The additional cost of this mechanism is that response message number is increased to $MN/2^l$, determined by the size of the network. Then the SRC of this mechanism is $O(2^{128}/MN)$. If fixed prefix is used in the network, the SRC will be $O(2^{64}/MN)$.

For a CGA address which requires the second hash value begins with k zeroes, the generation cost is $O(2^k)$, and the attack cost is $O(2^{62+k})$ [9]. Then is SRC of CGA is $O(2^{62})$.

The original DAD has an explicit SRC of $O(1)$. The relationship between the SCR is plotted in Figure 6. This figure shows that if the prefix is certain, the SCR of Pull DAD is close to CGA. If the prefixes are uncertain in the network, the SCR of Pull DAD is much higher than CGA.

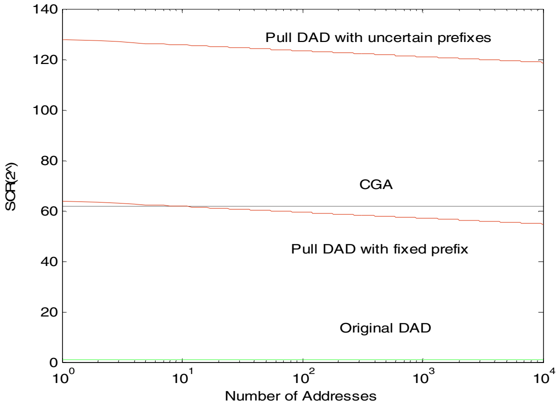

Figure 6. SRC Comparison

VI. Conclusion

In this article, we proposed a novel mechanism to secure DAD in IPv6 network. This mechanism uses pull model to avoid direct attack against the tentative address. This mechanism can also improve the security of address resolution and unreachability detection. Through evaluation, it is found to be a feasible and cost effective mechanism. Our future work is to optimize the hash function to enable more features in this mechanism.

Acknowledgment

Thanks to the support of China Science and Technology Supporting Project (No.2008BAH37B02), the National Basic Research Program (973) of China (No. 2009CB320501), the Middle School Student Soaring Project of Beijing City, the Specialized Research Fund for the Doctoral Program of Higher Education of China under Grant No.20090002110026, and CNGI project.

References

[1] S. Thomson, T. Narten, T. Jinmei, “IPv6 Stateless Address Autoconfiguration”, RFC4862, September 2007.

[2] J. Arkko, J. Kempf, B. Zill, P. Nikander, “SEcure Neighbor Discovery (SEND)”, RFC3971, March 2005.

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

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

[5] Zooko Wilcox-O'Hearn, “Names: Decentralized, Secure, Human-Meaningful:Choose Two”, http://shoestringfoundation.org/~bauerm/names/distnames.html.

[6] Christian Vogt, “Source Address Validation Improvement Protocol Framework”, http://tools.ietf.org/id/draft-vogt-savi-framework-01.txt.

[7] Zhenhai Duan, Kartik Gopalan, Yingfei Dong, “Push vs. Pull: Implications of Protocol Design on Controlling UnwantedTraffic”, in Proceedings of SRUTI ’05.

[8] G. Brassard, D. Chaum, and C. Cr´epeau. Minimum disclosure proofs of knowledge. *JCSS*, 37:156–189, 1988.

[9] Tuomas Aura, “Cryptographically Generated Addresses (CGA)”, ISC 2003.