International Journal of Future Generation Communication and Networking
Vol. 8, No. 4 (2015), pp. 193-202
http://dx.doi.org/10.14257/ijfgcn.2015.8.4.19

A Reverse Address Resolution Process with Variable Length Prefix

Song Guangjia[1], Ji Zhenzhou[1] and Wang Hui[2]

[1]*School of Computer Science and Technology, Harbin Institute of Technology, Harbin 150001, China*
[2]*National Computer Network Emergency Response Technical Team/Coordination Center of China, Beijing 100029, China*
35059899@qq.com, jizhenzhou@hit.edu.cn, 63211410@qq.com

Abstract

Address resolution is an important process in network communications. The primary function of address resolution is to determine the correspondence of a target network address to a physical address. The traditional address resolution process assumes that all the nodes on a network are honest and credible, and these nodes directly broadcast the resolution target address on the network. This process enables malicious nodes to easily mount attacks. We propose a reverse address resolution process with variable length prefix (called Re-AR) that obviates such attacks. According to the revelation principle, a node's <IP, MAC> mapping can be viewed as a private type in the address resolution process. In our proposed process, after a node receives an address resolution request broadcast, it unicasts the private type to a positive resolve node that assigns the communication to the correct node according to a predetermined mechanism. Based on simulation results, when the destination address is hidden in the broadcast packets, malicious nodes cannot easily conduct spoofing attacks according to the destination addresses. This phenomenon effectively prevents spoofing and significantly reduces the pollution rate of address cache tables.

Keywords: *Network Security, Address Resolution, Neighbor Discovery, Mechanism Design, Incentive Compatibility*

1. Introduction

One of the major functions of a network is to exchange information, with packets flowing from one node to another, routed by routers or delivered by switches. Networks use two methods to deliver packets: direct delivery and indirect delivery. If two communicating parties are on the same local area network (LAN), switches can deliver the packets by searching a <Port, MAC> mapping table, and the switch can deliver the packets to the target node through the port directly (i.e., direct delivery). In indirect delivery, both sides of the communications are on different LANs; therefore, the packets need to be forwarded by routers until they reach the target node's local network, after which they are sent to the target node by local switches using direct delivery. However, regardless of the kind of delivery, the target's MAC address must first be known before the packets can be delivered to the target. The process by which the target's MAC address is resolved from its IP address is called address resolution. This process is primarily conducted via the Address Resolution Protocol (ARP) in IPv4 or Neighbor Discovery Protocol (NDP) in IPv6. The purpose of address resolution is to determine the correspondence of the target node's network address to its physical address and to ensure that the address does not conflict with that of other nodes. The address resolution process used in IPv4 and IPv6 are fundamentally the same. First, node *A* sends an address resolution broadcast request, which requires any node that has the corresponding IP_X to

ISSN: 2233-7857 IJFGCN

reply with its MAC address—all nodes on the same link listen. If a node *B* has the corresponding IP_X, *B* replies with a packet that includes the <IP_X, MAC_B> mapping to node *A*, whereas other nodes remain silent. However, not all the nodes on a network are honest and credible. Malicious nodes always exist; therefore, this resolution process is vulnerable, with research showing that most attacks against the address resolution process evolve from spoofing.

Current research in the area of address resolution focuses on three aspects. The first is prevention and detection technology, which uses technical means to defend against or detect the attacks that have occurred. In general, this technique requires long-term monitoring and recording of every node's <IP, MAC> mapping. Consequently, if a node sends an address resolution packet whose <IP, MAC> mapping does not match those in the records, it is considered a deception [1, 2]. <IP, MAC> binding and VLAN division also belong to this category. However, this method is a passive defense, and it also increases the complexity of the network and maintenance costs. The second aspect is to enhance security by changing the protocol process. For example, in [3], a DHCP server is added to the LAN, which expands the DHCP protocol to complete the address resolution process. However, this method increases network costs and presents a single point of failure. The third aspect is encryption technology, such as using asymmetric cryptography to encrypt the ARP packets to prevent IP address theft [4]. To increase the security of NDP, the Internet Engineering Task Force (IETF) proposed SEcurity Neighbor Discovery (SEND), which uses Cryptographically Generated Addresses (CGA) and digital signature technology to encrypt communications, thereby effectively preventing IP address theft, as a resolution method [5, 6]. However, CGA requires significant amounts of centralized computing, and the cryptographically generated address still requires Duplicate Address Detection (DAD). When a conflict is discovered, the security level is increased and CGA is recalculated. This process may have to be performed 65,534 times in each instance; however, with Sec bit = 3, the computing time required will be measured in units of years, which is clearly intolerable. Therefore, reduction of the CGA computation time is essential. Parallel algorithms [7, 8] and faster encryption algorithms [9, 10] can effectively reduce this computation time. In addition, decreasing the given time condition to determine the Sec bit value is another effective method [11]. Meanwhile, the DAD attack problem still exists in the SEND protocol [12]. Although the security of the SEND protocol is strong, this protocol is difficult to implement, or only partially implemented [13]. Furthermore, coexistence of SEND with the NDP will generate new routing problems [14].

2. Address Resolution and the Wallet Problem

With developments and progress in human society, the Internet has become a public facility in most countries. However, a few decades ago, it was a proprietary facility and only provided services for the military, government, and universities. Nowadays, people can access the Internet anytime, anywhere, both wired and wirelessly, and can exchange information conveniently and access Internet resources rapidly. The demography of users of the Internet has also changed radically, from the original scientists, teachers, and other professionals to ordinary people. This phenomenon has resulted in changes from a small-scale trusted network to a large-scale untrusted Internet. From a sociological perspective, the Internet is an extension of human senses; thus, these changes in the user groupings mean that the network has higher security requirements.

Consequently, the problem of security should be considered first in any protocol design. Theoretically, two fundamental reasons are used for address resolution vulnerability. The first is that all the nodes in a network are assumed to be credible,

which is not the case; the second is that important information is broadcast on the network. These reasons are convenient for deceptive nodes.

In a network, a node's behavior reflects the programmer's behavior in some ways. First, from a security point of view, nodes in the network can be divided into three categories. The first category is reliability node, which includes routers, switches, firewalls, and servers, with features such as high security measures. The second category is the normal node, which has not been infected with malicious codes. The third category is nodes that are infected with malicious codes. First, we provide two definitions for network attack:

Definition 1: Incentive Compatibility (IC)—for the attacker, the utility of conducting an attack is greater than the maximum expected utility of not conducting the attack (normal participation).

Definition 2: Individual Rationality (IR)—for the attacker, the expected utility of conducting the attack is not more than the maximum utility of not conducting the attack.

Consider a situation that we call the wallet problem. In this scenario, the police have found a wallet that contains some cash x and identification (ID). How can the wallet be returned to its owner? There are two options.

Mechanism 1: First, an announcement can be issued stating that "The police have found a wallet, containing cash x and ID z. The owner is welcome to claim it; the rule is that the wallet will be given to the person who claims first."

Mechanism 2: First, issue an announcement stating that "I have found a wallet. The owner is welcome to claim it. However, you will have to answer a question: if you correctly describe the items inside the wallet (the amount of cash and ID information), then you can take the wallet, otherwise you might be given a negative record (for issuing a false claim).

Which mechanism would you choose? Most people would choose the latter, because the first is obviously unreasonable. Handing the wallet to the person who first claims it will only cause more people to become dishonest.

The wallet problem can be viewed as the three-stage game shown in Figure 1. In the first stage, A (the police) designs a mechanism and sends a signal. In the second stage, B and C (owner and impersonator) choose whether to accept the mechanism. If the mechanism is accepted, then, in the third stage, the game is started according to predetermined rules in the mechanisms.

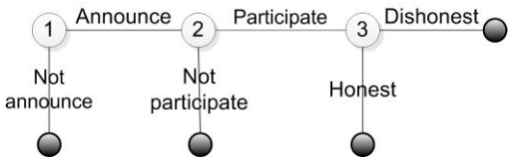

Figure 1. Three Stages in the Wallet Problem Game

In this three-stage game, if C chooses not to participate in stage 2, C obtains the exogenous reserve r. If C successfully claims the wallet, the utility is x. The claim process takes a toll y. If C's false claim is discovered by A, C will be punished, and C's utility becomes g. If A gives the wallet to the impersonator (C), A needs to provide compensation to the true owner, so the utility is $-x$. If the wallet is returned to the true owner, A receives bonus z. The probability that C successfully falsely claims is p, and the probability of failure is $1-p$. The expected utility (Eu) of C is

$$\mathrm{Eu}_C = p \cdot x + (1-p) \cdot g$$

The IR of C (IR_C) is

$$p \cdot x + (1-p) \cdot g > r$$

The expected utility of A is

$$Eu_A = p \cdot (-x) + (1-p) \cdot z$$

The IR of A (IR_A) is

$$p \cdot (-x) + (1-p) \cdot z > 0$$

To facilitate analysis, we assume that the toll is $0.01x$ and the bonus is $0.2x$. One owner and one impersonator are involved in the claim process. Assuming that no punishment is meted out to the impersonator, then $g = 0$. *C*'s exogenous reservation is a toll.

In mechanism 1, owner and impersonator have the same probability of obtaining the wallet, so for *B* and *C*, $p = 0.5$. *C*'s expected utility is:

$$Eu_C = 0.5x > 0.01x$$

Meet the IR. The expected utility of *A* is

$$Eu_A = 0.5 \cdot (-x) + 0.5 \cdot (0.2x) = -0.4x$$

Therefore, $Eu_A < 0$ does not satisfy the IR; if *A* is rational, it may choose not to send a signal. In mechanism 2, given that *A* will conduct an inquiry and *C* does not know how much money is in the wallet, *C* can only guess. However, the success rate of *C* is very low. Even if *A* relaxes the conditions and informs *B* the range of *x* is an integer from 1 to 100, the probability of *C* being successful is 0.01. Thus, *C*'s expected utility is

$$Eu_C = 0.01x + (-0.01x) = 0$$

Therefore, Eu_C is lower than the toll $0.01x$. *A*'s expected utility is

$$Eu_A = 0.99 \cdot (0.2x) + 0.01 \cdot (-x) = 0.197x$$

Thus, *A*'s utility approaches 20% of *x* virtually every time. Considering that IR_C is not satisfied if *C* is rational, *C* would choose not to participate in stage 2. The game tree of mechanism 2 is shown in Figure 2.

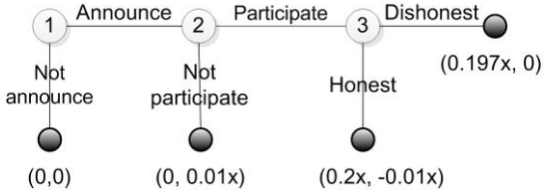

Figure 2. Three Stages of the Game at Mechanism 2

3. Reverse Address Resolution (Re-AR) with Variable Length Prefix

3.1. Re-AR Process with Variable Length Prefix

The address resolution process is similar to the wallet problem—it is also a three-stage game. The positive resolver can be viewed as police (*A*), the attacker can be regarded as the impersonator (*C*), and the passive resolver can be regarded as the owner (*B*). The mechanisms in the current address resolution protocols (ARP and NDP) are similar to mechanism 1 in the wallet problem; first, broadcasting the resolve destination address in the LAN, then giving the communication right to the node that replies first. This mechanism allows malicious *C* to easily attack at a very low cost. If the wallet has been given to an impersonator, the subsequent encrypted communication becomes meaningless.

IPv6 uses NDP to conduct address resolution. The format of an NDP message is depicted in Figure 3. First, suppose there are three hosts, namely, host *A*, host B, and host *C*. The IP configuration mode of LAN is Stateless Address Auto Configuration (SLAAC). In SLAAC, a node can configure its own link-local IPv6 address according to its MAC based on the EUI-64 rules. The details of the three hosts' IP and MAC information are given in Table 1. The address resolution process in NDP is as follows: when host *A* wants to send packets to *B*, if *A* does not have *B*'s MAC, *A* needs to broadcast NS to require *B* to reply with its MAC. Details of the NS packet are shown in Figure 4(a). In NS, the "Target address" field writes host *B*'s IP address. Every host in the LAN can receive this NS broadcast; if host *B* is on the link, *B* replies with an NA to *A*. Details of the NA are given in Figure 4(b). The "S" bit in the "Flags" field is set to one, which means that the NA is a reply to NS; the "option" field writes its own MAC address. When *A* receives this reply, *A* can extract *B*'s MAC from it to complete the address resolution process, and other hosts remain silent [15].

Table 1. IP and MAC Information of Three Hosts

Host	IP	MAC
A	FE80::A8AA:AAFF:AAFE:AAAA	AA:AA:AA:AA:AA:AA
B	FE80::B9BB::BBFF:FEBB:BBBB	BB:BB:BB:BB:BB:BB
C	FE80::CECC:CCFF:FECC:CCCC	CC:CC:CC:CC:CC:CC

Ethernet Header	Dest MAC Src MAC Type
IPv6 Header	SRC IP DST IP Next header
IPv6 Data	Flags Type Target address option

Figure 3. Format of NDP Message

NS

Ethernet Header	33:33:FF:BB:BB:BB AA:AA:AA:AA:AA:AA 0x0806
IPv6 Header	FE80::A8AA:AAFF:FEAA:AAAA FF02::1:FFBB:BB:BB 0x3A
IPv6 Data	135 FE80::B9BB:BBFF:FEBB:BBBB AA:AA:AA:AA:AA:AA

(a)

NA

Ethernet Header	AA:AA:AA:AA:AA:AA BB:BB:BB:BB:BB:BB 0x0806
IPv6 Header	FE80::B9BB:BBFF:FEBB:BBBB FE80::A8AA:AAFF:FEAA:AAAA 0x3A
IPv6 Data	136 S=1,O=1 FE80::B9BB:BBFF:FEBB:BBBB BB:BB:BB:BB:BB:BB

(b)

Figure 4. (a) NS and (b) NA

In the address resolution process, a node's <IP, MAC> mapping can be observed as a private type; no one knows the private types of others. *A* conducts address resolution to allow *B* to obtain the communication right. In the address resolution process, normal nodes are typically inclined to give real IP and MAC information; the malicious nodes tend to use false information to deceive, thereby obtaining more utility. To motivate all the nodes to show their true types, according to the revelation principle [17], we use a direct mechanism. In stage 1, *A* designs the

mechanism and sends a signal. In stage 2, all the nodes accept the mechanism. In stage 3, all the nodes give their true types (IP and MAC information), then *A* gives the communication directly to the specific node in accordance with the predetermined mechanism. We call this process Re-AR.

The message format used in Re-AR is shown in Figure 5. The format is similar to that of an NDP message, with one major difference, the "Prefix" field. In Re-AR, the address resolution process is as follows. First, *A* performs a Re-AR request broadcast; the packet structure is shown in Figure 5. Unlike the NS in IPv6, the "Target address" field here is empty, and the "Prefix" field is used to give out the prefix information of the destination address. The advantage is that when other nodes receive the Re-AR request, they cannot see the resolution destination address from the broadcast, but they will know the prefix of the address being resolved according to the prefix information in the request. This feature plays a crucial role in the process of information hiding. If matching addresses exist, then nodes will reply with a Re-AR reply to *A* by unicast. Within the given time, *A* checks all the reply packets it receives and, if a packet whose "Target address" field matches the destination address, then the address resolution is successful; otherwise, the address resolution process has failed. A flowchart of the process used by Re-AR is shown in Figure 6.

Ethernet Header	Dest MAC Src MAC Type
IPv6 Header	SRC IP DST IP Next header
IPv6 Data	Flags Prefix Type Target address option

Figure 5. Message Format Used by Re-AR

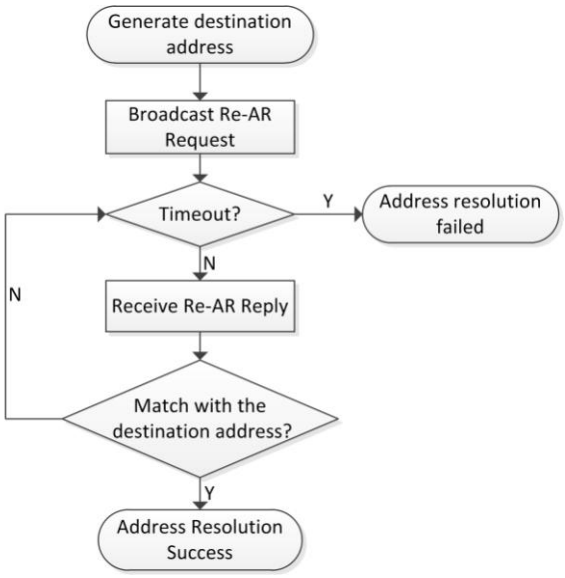

Figure 6. Flowchart of the Process Used by Re-AR

In Re-AR, when host *A* needs to communicate with host *B*, *A* first checks its own address cache and, if *B*'s MAC is not there, *A* performs address resolution. First, *A* performs a request broadcast, with the request and reply shown in Figure 7. Note that in this request, the "Target address" field is set to "::", not filled with the address resolution

destination address. This measure is conducted to hide the destination address. When another host receives this broadcast, it checks the "Prefix" field and, if the address is in compliance with the "Prefix," it unicasts a reply to host *A*; if no match is found, the host simply drops the request packet. In Figure 7(a), the "Prefix" is filled with "FE80," indicating that the destination is a link-local address, so host *B* needs to reply to it. Details of the reply are shown in Figure 7(b). Within a certain time, *A* checks all the replies and, if a reply is found in which the "Target address" field matches with the resolution destination address, then address resolution is successful.

Request

Ethernet Header	33:33:FF:BB:BB:BB AA:AA:AA:AA:AA:AA 0x0806
IPv6 Header	FE80::A8AA:AAFF:FEAA:AAAA :: 0x3A
IPv6 Data	135 FE80 FE80:B9BB:BBFF:FEBB:BBBB AA:AA:AA:AA:AA:AA

(a)

Reply

Ethernet Header	AA:AA:AA:AA:AA:AA BB:BB:BB:BB:BB:BB 0x0806
IPv6 Header	FE80:B9BB:BBFF:FEBB:BBBB FE80::A8AA:AAFF:FEAA:AAAA 0x3A
IPv6 Data	136 S=1,O=1 FE80 FE80:B9BB:BBFF:FEBB:BBBBB B:BB:BB:BB:BB:BB

(b)

Figure 7. (a) Request and (b) Reply used in Re-AR

3.2. Security Analysis

Let us now consider the definition of Address Resolution Hit Rate (ARHR). In the address resolution process, ARHR is the ratio of the number of times the resolve destination address exists in the LAN to the total number of times address resolution is conducted.

ARHR = (# of times destination address exists in LAN)/(# of times address resolution conducted)

In the address resolution game, the player set is *N*, where $N = \{A, B, C\}$. *A* represents the positive resolver, *B* represents the passive resolver, and *C* represents the attacker. Their focus is on the allocation of communication rights. If *C* obtains the right communication, the utility of *C* is r_2; otherwise, the utility is zero. If *A* assigns the communication directly to *B*, then the utility of *A* is r_1; otherwise, A's utility is $(-r_1)$. In stage 2, *C* chooses to be involved, because if *C* chooses not to become involved, *C* will not have any chance to get the communication right. In stage 3, S is the strategy set of *C*, where $S = \{s_1, s_2\}$; s_1 is the use of random address spoofing to get the communication right, and s_2 is the reply with the true $<IP_C, MAC_C>$ mapping. The strategy set of *B* is S_B, which includes one strategy, so $S_B = \{s^*\}$. We know that address resolution is often initiated by the upper layer protocols. In the case of the server area in our university, ARHR is approximately 35%. Suppose in the address resolution process, the length of the prefix in NS is 64 bits, 2^8 nodes exist in the LAN, and *C* replies with random address (strategy s_1) and true <IP, MAC> information (strategy s_2), the resulting utilities of *A* and *C* are shown below:

The utilities of *A* are

$$u_A(s_1,s^*) = 0.35r_1+(-r_1/2^{64})\approx 0.35r_1$$

$$u_A(s_2,s^*) = 0.35r_1$$

The utilities of *C* are

$$u_C(s_1,s^*) = r_2/2^{64}$$

$$u_A(s_2,s^*) = (1/2^8) \cdot 0.35 \cdot r_2 \approx 0.00137 r_2$$

For *C*, using real addresses to reply will clearly gain more revenue. Thus, in this game, showing the true type is the dominant strategy for all the nodes to satisfy the IC. The game tree of Re-AR is shown in Figure 8.

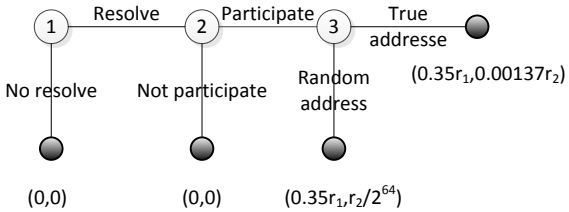

Figure 8. Re-AR Game Tree

4. Simulation Experiments

To verify the efficacy of Re-AR, we conducted simulation experiments in which we used the OPNET simulation software with a star network topology. The network comprises a switching node, with eight traffic nodes surrounding it. Each node contains two packet sources: the address resolution packet source and traffic packet source. To simulate a real environment's traffic data and address resolution, data statistics from a university core switch (Quidway S9306) are used. The statistics show that address resolution packets account for approximately five percent of traffic packets. One experimental statistic is the ratio of Re-AR traffic to the total traffic; the other is the address cache pollution rate (the ratio of error entry to total entry).

In terms of traffic, the traditional address resolution process uses a broadcast and a hit node response; traffic is *n+1*. Re-AR needs a broadcast and a collection of responses; the traffic is *n+m*, where $m \leq n$. Hence, Re-AR's traffic is twice that of the traditional protocol. Our experimental results show that Re-AR address resolution packets account for approximately nine percent of the total traffic, whereas in NDP, the packets account for five percent; the details are shown in Figure 9. In another part of the experiment, we simulated a scenario in which attack nodes exist and observed how the address cache pollution rate changed. Once the address cache is contaminated, packets are sent to the wrong destination in subsequent communication processes, resulting in communication interruptions, such as packet loss. As shown in Figure 10, in an attack situation, Re-AR's cache pollution rate is much lower than that of the traditional protocol.

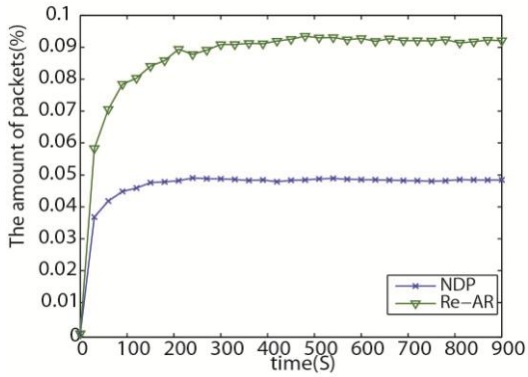

Figure 9. Comparison of NDP's and Re-AR's Percentage of Address Resolution Traffic

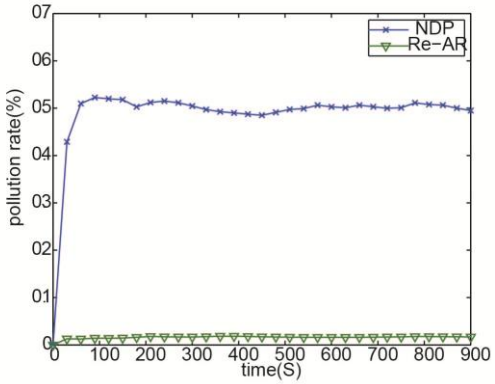

Figure 10. Comparison of NDP's and Re-AR's address cache pollution rate

5. Conclusions

Address resolution process plays an important role in the hierarchical network architecture. Attacks against the address resolution process are the main threats to the LAN security. The consequences caused by attacks against the address resolution process are very serious. In this letter, we analyzed the security of the address resolution process from the perspective of game theory. Node's <IP, MAC> mapping can be regarded as a private type; no one knows the private types of others. According to the revelation principle in mechanism design, we have proposed a direct mechanism called Re-AR. In Re-AR, whether it is the attack node or the normal node, giving the real IP and MAC in the reply is the dominant strategy for equilibrium to satisfy the IC constraint; therefore, Re-AR fulfills the purpose for which it was designed. If a node lies, its utility is reduced because lying is not a dominant strategy. Simulation results show that Re-AR can effectively reduce the contamination rate of the address cache, and it has stronger security than the conventional method.

Acknowledgements

This paper is supported by National Nature Foundation of China under Grant No. 61173024.

References

[1] S. Yeob, D. Kim and J. Kim, "Enhanced ARP: preventing ARP poisoning-based man-in-the-middle attacks", IEEE Communications Letters, vol.14, no.2, **(2010)**, pp.187 - 189.
[2] M. Oh, Y.-G. Kim, S. Hong and S. Cha, "ASA: Agent-based secure ARP cache management", Communications, IET, vol.6, no.7, **(2012)**, pp.685-693.
[3] B. Issac and L. A. Mohanmmed, "Secure unicast address resolution protocol (S-UARP) by extending DHCP", Proceedings of IEEE 13th International Conference on Networks Jointly held with 7th International Conference on Communication, **(2005)**; Koala Lumpur, Malaysia.
[4] V. Goyal and R. Tripathy, "An efficient solution to the ARP cache poisoning problem", Lecture Notes in Computer Science, vol.3574, **(2005)**, pp.40–51.
[5] J. Arkko, ED, J. Kempf, B. Zill and P. Nikander, "Secure Neighbor Discovery (SEND)", RFC3971, **(2005)**.
[6] T. Aura, "Cryptographically Generated Addresses (CGA)", RFC3972, **(2005)**.
[7] A. AlSa'deh and C. Meinel, "Secure Neighbor Discovery: Review, Challenges, Perspectives, and Recommendations", IEEE Security Privacy, vol.10, no.4, **(2012)**, pp.26-34.
[8] Hosnieh, Rafiee, A. AlSa'deh and C. Meinel, "Multicore-Based Auto-Scaling Secure Neighbor Discovery for Windows Operating Systems", Proceedings of IEEE International Conference on Information Networking (ICON), **(2012)**; Bali.
[9] A. Alsa'deh and F. Cheng, "CS-CGA: Compact and more Secure CGA", Proceedings of IEEE International Conference on Networks, **(2011)**; Singapore.

[10] Qadir, Sana and M. U. Siddiqi, "Cryptographically Generated Addresses (CGAs): A survey and an analysis of performance for use in mobile environment", Proceedings of IJCSNS International Journal of Computer Science and Network Security, vol.11, no.2, **(2011)**.
[11] A. AlSa'deh, H. Rafiee and C. Meinel, "Stopping Time Condition for Practical IPv6 Cryptographically Generated Addresses", Proceedings of IEEE International Conference on Information Networking (ICON), **(2012)**; Bali.
[12] S. Guangxue, W. Wendong and G. Xiangyang, "A quick CGA generation method", Proceedings of IEEE International Conference on Future Computer and Communication, **(2010)**; Wuhan.
[13] H. Rafiee, A, AlSa'deh and C. Meinel, "Winsend: Windows Secure Neighbor Discovery", Proceedings of ACM International Conference on Security of Information and Networks, **(2011)**; Sydney, Australia.
[14] H. Yi, W. Zhenxing, W. Yu and Z. Liangchen, "Routing Attack in the ND and SEND Mixed Environment", Proceedings of IEEE International Conference on Multimedia Information Networking and Security, **(2012)**; Nanjing, China.
[15] T. Narten, W. A. Simpson, E. Nordmark and H. Soliman, "Neighbor discovery for IP version 6 (IPv6)", **(2007)**.

Authors

Song Guangjia, he is currently a Ph.D. candidate at Harbin Institute of Technology, Harbin, China. His research interests include network security, address resolution protocol, academic credibility.

Ji Zhenzhou, he received the B.E., M.S. and Ph.D. degrees in Computer science and technology from Harbin Institute of Technology, Harbin, China, he is currently a professor and Ph.D. supervisor in Harbin Institute of Technology. His research interests include advanced computer architecture, parallel computing technology, computer network security and the QoS system.

Wang Hui, he received Ph.D. degrees in Computer science and technology from Harbin Institute of Technology, Harbin, China, he is currently an engineer in National Computer Network Emergency Response Technical Team/Coordination Center of China. His research interests include satellite network, network modeling and network security.

