A Novel SeND Based Source Address Validation Mechanism (SAVM-SeND)

Pingping Lin  
Network Architecture and IPv6 Lab  
Tsinghua University  
Tsinghua National Laboratory for Information Science and Technology  
Beijing, China, 100084  
linpp@netarchlab.tsinghua.edu.cn

Jun Bi  
Network Architecture and IPv6 Lab  
Tsinghua University  
Tsinghua National Laboratory for Information Science and Technology  
Beijing, China, 100084  
junbi@cernet.edu.cn

Abstract—SAVM-SeND, a solution to provide source address validation using the SeND protocol [1], is proposed in this paper. It assumes all the hosts are configured with SeND protocol in IPv6 networks and aims to provide a mechanism for the control of the source addresses spoofing in all of the address assigning circumstances. Besides, it supports each host with multiple interfaces, each interface with multiple MAC addresses, port mobility, and can achieve a host level granularity filtering in the access network. In the end, the evaluation is discussed.

Keywords-SeND protocol; public key; source address validation

I. INTRODUCTION

In the current Internet, attackers can easily use deliberately or randomly generated source addresses to generate packets and launch attacks. However, the existed solutions to this problem are either cannot solve this problem in an appropriate granularity or only effective to some certain network environments. So, the SAVM-SeND is proposed for IPv6 networks configured with SeND protocol.

The rest of this paper is organized as follows: Section II and III introduce the related work and design considerations respectively. SAVM-SeND is described in section VI. Then section V will analyzes how the solution deals with some special cases. Section VI gives the evaluation. And section VII concludes this paper.

II. RELATED WORK

There are several solutions working in the access network. IP source guard function is a solution running on layer 2 devices. It uses the binding among IP address, MAC address, and port number of switch to filter packets. However, when one switch port connects to multiple hosts, like with hubs, hosts under a same port can forge each other’s addresses. Besides, if attackers use attack tools to generate packets with the source address of another host B in the same hub, host B could not notice the address collision. In IEEE 802.1X extension protocol, once one host under a hub passed the authentication, the related physical port of switch will be opened, and then the other hosts under the same hub can also connect to the Internet without do the authentication. Ingress filtering [2] can’t achieve a host level granularity filtering. CSA [3] needs to change the protocol stack on hosts.

The proposed solution in this paper aims to provide a mechanism for the control of the source addresses spoofing in all of the address assigning circumstances with no host change. Besides, this solution supports each host with multiple interfaces, each interface with multiple MAC addresses, and port mobility, and can achieve a host level granularity filtering.

III. DESIGN CONSIDERATIONS

A. Scope of SAVM-SeND

If MAC address is secure which means the MAC address cannot be forged by others, the SAVM-SeND almost can solve all the cases, including one host with multiple interfaces, one interface with multiple addresses, ports of switch or router connected to hubs, changing port, and no matter what the environment is: wire or wireless. And in this condition, we can take the MAC address as the binding anchor. However, if the MAC security cannot be guaranteed, it can’t solve the following situations: ports connected to hubs and the wireless scenarios.

B. Constraints for SAVM-SeND

The SAVM-SeND relies on the usage of SeND protocol and the CGA [5] with no changes to hosts.

C. Application Environments

The SAVM-SeND function takes effect in forwarding devices which are directly connected to hosts, like switches.
IV. SAVM-SeND SPECIFICATION

A. SAVM-SeND data structures

In different IP addresses assigning circumstances: stateless, static, DHCP servers, SeND/CGA, the work steps of SAVM-SeND are different. But in common, the forwarding devices maintain a binding table which stores the mapping among IP, MAC, public key, port, lease time and other attributes.

1) DHCP server or static address assignment: in this scenario, the mapping item in the table is as follows:

<IP_A, MAC_A, Pubkey_A, T_Leasetime, port, others>

2) SeND/CGA or stateless address auto configuration: in this scenario the mapping item in the table is as follows:

<IP_A, MAC_A, Pubkey_A, port, others>

B. SAVM-SeND with MAC security

1) Stateless address auto configuration:

Assume host A is configured with MAC_A and IP_A. Once the switch receives a packet with IP_A and MAC_A, it will do a lookup to find if any related item exists in its binding table. Because this is a new pair of addresses, it won’t find any corresponding item. Then, the switch will forward this packet, and start up a DAD [1] for IP_A immediately. After the host A received the NS [4] packet for DAD, it will return a NA [4] packet, which takes the host A’s public key A and signature. Next the switch will carry out a validation for the CGA and RSA signature options for NA packet. If this packet passes the validation, the switch will extracts the public key A from this packet, generates a mapping item with MAC_A, IP_A, public key A, the corresponding switch port, and inserts this item into its binding table. The format of this item is like this:

<IP_A, MAC_A, Pubkey_A, port, others>

As described, the main process can be expressed as follows:

In the data communication process, since MAC addresses are secure, the switch only needs to check the mapping relationship between MAC addresses and IP addresses. Then why should the solution need to bind host A’s public key? In this solution, the public key is mainly used to deal with the following situations: each host with multiple interfaces, each interface has multiple addresses, and port mobility. Later, this paper will give a detail explanation for this in the section how SAVM-SeND handles special cases.

2) DHCP server assigning addresses:

In this situation, the switch must enable its snooping function firstly. During the progress of host A’s requesting IP address from its DHCP server, once the switch received the DHCPOFFER packet from the DHCP server, It will extracts the IP_A and MAC_A, the lease time and other information if necessary. Then the switch binds the IP_A, MAC_A, lease time, and switch port as one item, and inserts this item into its binding table. So far, the content of this item is as follows:

<IP_A, MAC_A, T_Leasetime, port, others>

Each time the switch inserts a new item, it must launch a DAD to get a corresponding public key. When host A receives the DAD packet, it will reply a NA packet including the public key and the signature of host A. Then, if the CGA and signature passes the validation by switch, the switch will add the public key to the item above. Thus, until now that item has changed to the following format:

<IP_A, MAC_A, Pubkey_A, T_Leasetime, port, others>

In figure 2, for the switch each interface equals one host. If the two different MAC addresses want to use different IP addresses, then each MAC can be mapped to a public key. Only when different MAC addresses want to use a same IP address, should the host return a same public key in NA.

3) Static address assignment

After a host has being assigned one available IP address, the following steps are the same as the situation after get a
IP address from a DHCP server in the situation of DHCP server assigning addresses. But there are some differences in changing port, and these will be described in detail in the special cases later.

4) **SeND/CGA**

The CGA is generated from the public key, so the CGA combining the signature can also prove the ownership of a message, so the CGA have the ability to take the place of the public key. However, in order to achieve uniform and easy management, in this scenario, the SAVM-SeND should also add the public key to the mapping item as follows:

\[
<\text{CGA}_A, \text{MAC}_A, \text{Pub}_{keyA}, \text{port}, \text{others}> 
\]

So, for the host has multiple interfaces, each interface has multiple MAC addresses, the binding solution is the same as described in the stateless situation, the only difference is that the binding IP is the CGA rather than the normal IP address.

And then in the data communication process, the switch only needs to check the mapping relationship between MAC address and CGA since MAC addresses are secure.

**C. SAVM-SeND when MAC security is not guaranteed**

1) **The binding anchors**

When MAC address is secure, the solution can take MAC addresses as binding anchors.

But when MAC security is not assured? Here, this paper only considers the port exclusive scenario. In this scenario, the attributes in one mapping item are the same as those in MAC security scenarios. The only difference is which attribute can play the role of anchor. When the MAC security is not assured, the binding can use the switch port as its anchor.

2) **SAVM-SeND solution**

The process of generating the binding table is the same with that in the MAC security scenarios. But, when the MAC addresses are secure, the switch can check each packet by the mapping of MAC address and IP address. When the MAC addresses are not secure, switch should check each packet by the mapping of switch port, IP address and MAC address.

**V. HANDLING SPECIAL CASES**

A. **Multiple MAC addresses use a same IP address**

In this case, take the situation of DHCP server assigning IP addresses for example, in other address assigning situations, the process is the same.

There is a special scenario which must be put into consideration. At first, the host A’s MAC$_A1$ was connected to port1 in the switch, and use the IP$_A$. Then, a second MAC$_A2$ of host A also wants to connect to the port2 in that switch, and also want to use the IP$_A$. Then how should the switch differentiate whether the MAC$_A2$ belongs to host A or belongs to another host like B? Because if the MAC$_A2$ belongs to host A, the switch will allow it to use the IP$_A$, but if the MAC$_A2$ belongs to host B, the switch must forbid it to use the IP$_A$. Because this is address spoofing.

The content mentioned above has already described the mapping item among the public key, IP address and MAC address. Here it assumes host A have already established a binding item for MAC$_A1$ as follows:

\[
<\text{IP}_A, \text{MAC}_A, \text{Pub}_{keyA}, \text{Leasetime}, \text{port}, \text{others}> 
\]

Thus, once the first packet with MAC$_A2$ and IP$_A$ arrives, the switch won’t find a mapping item for the MAC$_A2$ and IP$_A$ in its binding table. In order to distinguish this address pair is from host A or from an attacker, the switch should launch a DAD for the IP$_A$ immediately. Thus, only if the responding NA packet takes the public key A, and the right signature, the switch can conclude MAC$_A2$ and MAC$_A1$ belong to a same host A. And then, it permits the mapping between MAC$_A2$ and the IP$_A$, and inserts this new mapping item into the binding table, until now the related items are as follows:

\[
<\text{IP}_A, \text{MAC}_A, \text{Pub}_{keyA}, \text{Leasetime}, \text{port}, \text{others}> 
\]

\[
<\text{IP}_A, \text{MAC}_A2, \text{Pub}_{keyA}, \text{Leasetime}, \text{port}, \text{others}> 
\]

Otherwise, if the signature or the public key is not right, the switch will think another host like host B may want to use or forge IP$_A$, so the switch will drop this kind of packets.

What’s more, in this special case, if the MAC$_X$ want to use a same address IP$_Y$ which has already been used by MAC$_Y$, it must return the same public key Y which is also used by MAC$_Y$ in NA packet in DAD.
B. Multiple MAC addresses use different IP addresses respectively

In this case, as we discussed in figure 2, from the perspective of the switch, the two situations are the same in handling. So, the switch can also treat each interface as one independent host.

C. One interface with multiple MAC addresses

Assume an interface of host A has multiple MAC addresses, and also has multiple IP addresses. As long as IP and MAC addresses are from the same interface, they can combine with each other arbitrarily as follows:

\[ IP_A | MAC_A | Pub_key_A | IP_{others} \]

D. Port mobility

In stateless, SeND/CGA, DHCP server address assigning situations, once the switch find a link downs, it should delete all the items related to that host. In wireless environment, this may be achieved by a periodically reachable detection to its clients. For the static and manual, the related binding item shouldn’t be deleted once a link downs. And it should be decided by administrators.

VI. Evaluation

SAVM-SeND takes the public key as an identity for a MAC address or a host, with the public key it can automatically judge whether multiple MAC addresses intend to use a same IP address are from a same host. With no change to hosts it can be easily deployed. Further, In SeND packet, there are three IP addresses, Destination, Source and Target. One possible address configuration is as follows. IPA is either a general IP address or a CGA.

Compared with IP source guard, SAVM-SeND has more communication steps, namely the DAD process actively launched by the switch. We can treat the DAD process as a communication in control plane. And in the process of data transmission, the work for switch to do is just filtering by its binding table. So, in this way, the performance of SAVM-SeND is equal to the IP source guard solution. But it supports each host with multiple interfaces and other complex situations.

VII. Conclusion

This paper discussed a novel source address validation mechanism for IPv6 networks configured with SeND protocol. The solution can deal with all the address assigning circumstances. And supports complex situations: each host with multiple interfaces, each interface with multiple MAC addresses, port mobility, and can achieve a host level granularity filtering. The key technique to achieve this is that the forwarding device makes full use of the SeND protocol and actively launches a DAD process to generate a mapping item.

Besides, this solution takes the public key as identity for a MAC address or host according to the specific network environment to achieve automatically identify whether two MAC addresses are from a same host or two different hosts. According to the features described for CGA in [5], though the public key is used in SAVM-SeND, it does not need any PKI infrastructure.

In the end, by analyzing and comparing with other solutions already in use, the SAVM-SeND is feasible.

REFERENCES