A link signature based DDoS attacker tracing algorithm under IPv6

Ting Ma
Department of Electronic and Information
Ningbo DahongyingUniversity
Ningbo, P.R.China
Email: happy_pepper@126.com

Abstract

The ipv6 security architecture, IPSec, plays a positive role in the protection of IPv6 networks. To some special attacks, especially DDoS attacks, IPSec appears relatively weak, because IPSec can only defend against DDoS attacks that spoof their source addresses. In cases where attackers launch DDoS attacks with their real identity, IPSec is helpless. This paper proposes a link signature based DDoS attacker tracing algorithm. It can immediately reconstruct the entire attack path after suffering a DDoS attack whether or not the source addresses are spoofed. To verify the validity of our algorithm, we implemented it under a simulated IPv6 environment with the OMNeT++ IPv6Suite.

Key words*: IPv6 DDoS attacker tracing, packet marking, link signature, OMNet++ IPv6Suite*

1. Introduction

With the development of internet, IPv6 will gradually replace IPv4 as the next generation internet protocol. Correspondingly, the focus of study in network security should also gradually change to IPv6 networks. The IETF proposed new network security architecture, IPSec, when formulating the IPv6 standards. IPSec plays a positive role in IPv6 network security, but to some special attacks, for example DDoS attacks, it looks relatively weak. IPSec is only able to protect hosts from DDoS attacks that spoof their source addresses. Where DDoS attacks are launched with their true identity or the victim has no IPSec configured, IPv6 networks are inevitably exposed to DDoS attacks. Because of this loophole, we propose a new signature based DDoS attacker tracing algorithm under IPv6 that helps network administrators actively and effectively treat DDoS attacks while underway. This algorithm can reconstruct the entire attack path in a short time when suffering a DDoS attack. In order to verify the validity of this algorithm, we implemented it under a simulated IPv6 environment using the simulation tool OMNeT++.

In Section 2 of this paper we briefly introduce the correlative background of our algorithm. Section 3 gives a detailed description of this algorithm. In Section 4 we present the process of our simulation. Finally we analyze the advantages and disadvantages of this algorithm in the concluding section.

2. Background

In this section we mainly concentrate on explaining the correlative background of our link signature based DDoS attacker tracing algorithm. We assume that the reader is familiar with the basic IPv6 protocol.

2.1. DDoS attacks under IPv6

DDoS means Distributed Denial of Service attack. In a DDoS attack, the attacker controls many puppet machines to simultaneously inundate the victim with network traffic, denying them continuity of service. There are two basic means to implement DDoS attacks. One is to generate large volumes of traffic to victim and the other is to send malformed packets. In both cases victims exhaust their resources processing these malicious packets to the extent that normal service cannot be maintained. These two methods can work under IPv6. We will mainly concentrate on the more frequently used traffic volume type attack in this paper.

Attacker tracing is a key technology in DDoS defense mechanisms, because the real attacker behind the DDoS attack often spoofs the source addresses. It is practically impossible to acquire the identity of an attacker in a direct way. An effective tracing algorithm that can provide the entire attack path is therefore required.

2.2. IP traceback technology

It is well known that IP is a stateless protocol, so we can not directly acquire any path information from received packets. Still worse, it allows senders to fill packets' source addresses themselves, allowing attackers to spoof their source address. For the above reasons it is almost impossible for the victim to directly find the real attacker when under attack. Considering the only information the victim can directly get is that in the received packets, researchers proposed IP traceback technology which is used to locate the attack according to information from received packets.

IP traceback technology has two intentions. One is to find the IP address of the attacker and the other is to reconstruct the whole attack path. Since attackers nearly always spoof their source address, the second of these is used to locate the DDoS attacker.

Current research into IP traceback technology includes import filter technology, link testing technology, the log technique, ICMP tracing technology and packet marking technology. Our link signature based DDoS attacker tracing algorithm is based on packet marking. This method is based on the two main ideas packet marking, to enable, path reconstruction. Every router marks attack packets when forwarding them and the victim reconstructs the attack path according to the marks on received attack packets. In this process the method of marking forwarded packets is called the packet marking algorithm and the used to reconstruct the attack path is called the reconstruction algorithm.

3. A Link signature based DDoS attacker tracing algorithm under IPv6

Based on the IP traceback technology, we propose the link signature based DDoS attacker tracing algorithm. According to the core thinking of packet marking technology, our tracing algorithm is mainly composed of a packet marking algorithm and a reconstruction algorithm. In this Section we give a detailed description of this algorithm by analyzing these two sub-algorithms.

3.1. The packet marking algorithm

There are many mature packet marking algorithms, including, AMS1, AMS2 [10]. We propose a relatively simple packet marking algorithm based on these two known algorithms. Firstly we assign each link in the network a unique signature that the router

adds to the attack packet's headers when forwarding it. This 'marking' involves XORing a packet's signature area with its out or in link's signature. In this paper we mark the attack packet according to the signature of the out link, that to which the packet is sent. Because each forwarding router marks the attack packet, when arriving at the victim the attack packet displays the whole attack path. We call the signature assigned for each link its 'link signature' and the XOR result of every link signature on a whole path the 'path signature'.

Figure 1 illustrates the process of the marking algorithm. As shown in figure 1, a packet was marked by Router1, Router3 and Router4 in-order when transferring from Host1 to Host2. The initial value of this packet's signature area is 0 when arriving at Router1. Router1 marks this packet according to its out link's link signature. After marking, the value of this signature area is changed to 48079. According to the shortest path first algorithm, Router3 and Router4 subsequently mark this packet. Finally, when this packet arrives at Host2, its signature area is finally changed to 28714. As mentioned earlier, this value is the whole path's path signature.

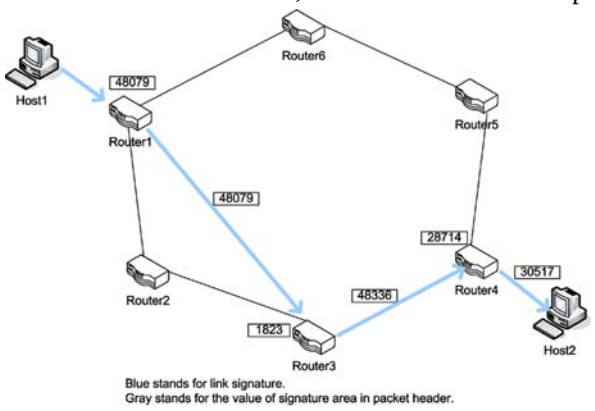

Figure 1. Example of packet marking process

The packet marking algorithm faces the following key issues:

1. How to assign link signatures

There are several methods to solve the problem of how to assign a unique signature to each link. For example, we can use hash or similar algorithms. In this paper, we assign 16-bit long link signatures randomly for each link. 16 bits is long enough to meet the tracing demand. Uniqueness is guaranteed by checking each newly assigned signature and randomly regenerating a replacement if it is not unique.

2. How to choose the signature area from the IPv6 header

We should choose an area from the IPv6 header to record the path signature of the attack. Through analysis we found that the flow label area of IPv6 header is not commonly used. It is mainly used to identify a series of packets which need special treatment. Given DDoS attacks are often disguised as the normal traffic we chose the flow label area as the signature area. We can also use the IPv6 extension headers to record the path signature where the flow label area is needed, for example, for the real-time or QoS services.

3.2. Reconstruction algorithm

The reconstruction algorithm is responsible for reconstructing the whole attack path according to the value of the signature area in received attack packets. We can reconstruct the attack path statically or dynamically. Dynamic path reconstruction

during a DDoS attack would increase the traffic burden on the victim and the dynamic process is also relative complex. As static reconstruction, relatively, does not suffer from either of these drawbacks, it is prefferable for the DDoS attacker tracing algorithm.

To function, the static construction process needs the entire network topology and the link signature of each network link. Static reconstruction calculates the tracing information for the shortest path from each host to the victim at the beginning of the process. This tracing information includes the hops, path signature and IP address of each passing router in the path. The path signature is calculated by XORing all link signatures on the path. During calculation, we can also record the addresses of all forwarding routers and the hops of this path. When the victim receives the marked attack packets all that needs to be done is to find the path with the same signature and hop values as the received packets. This provides the IP addresses of all routers in the path of the attack packet.

4. The simulation of a link signature based DDoS attacker tracing algorithm under IPv6

4.1. The OMNeT++ IPv6Suite

The IPv6Suite module in OMNeT++ was used to simulate our tracing algorithm. OMNeT++ is an object-oriented modular discrete event network simulator. It uses the NED language to describe network topology and C++ to program simple modules. Compared to the traditional simulation tool NS2 it has the advantages of clear structure and rich interface. High modularization makes it's unnecessary to re-compile the entire source code after modification. It also takes advantage, in simulating large networks, of its high operating speed compared to OPNeT. Further as it is open source, it is the first choice for academic research.

IPv6Suite is an open source OMNeT++ model suite for accurate simulation of IPv6 protocols and networks. It extends the INET Framework by adding models for the simulation of functionality as described in the following RFCs: RFC2373, RFC2460, RFC2461, RFC2462, RFC2463, RFC2472 and RFC2473.

4.2. Implementation of the proposed link signature based DDoS attacker tracing algorithm in an IPv6 simulation environment

Because OMNeT++ is a modularized simulator, we defined five functional modules to complete our simulating process. These five modules combine with each other by sending messages. Figure 2 illustrates the combination of these functional modules.

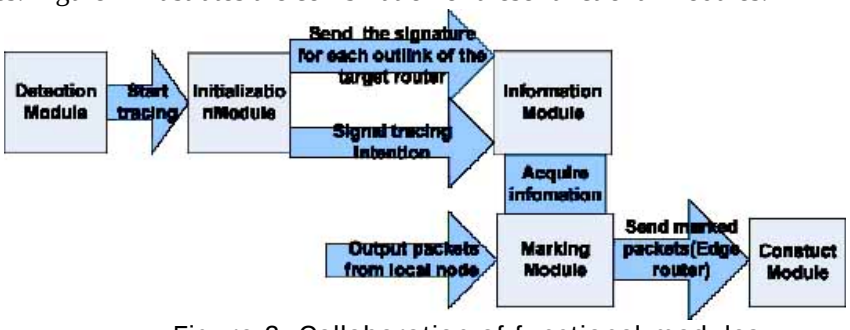

Figure 2. Collaboration of functional modules

We implemented these five modules by adding to or extending modules from IPv6Suite. The function and implementation of these modules is as follows.

1. Detection module

This module is used on the edge routers of the network. It is mainly used to detect the traffic to and from hosts linked to the edge router. Currently we almost determine DDoS attacks by the traffic profile of victim. Statistics show that DDoS victims' traffic bursts then follows a smooth trend during the attack. As we only focus on tracing DDoS attacks in this paper, we detect DDoS attacks in a simple way only considering the victim's average traffic levels. More precisely we periodically calculate the average traffic for the victim every 10 seconds. Once the average traffic exceeds a predefined threshold we consider the victim under DDoS attack.

We implemented the detection module by extending the IPv6Output module in the IPv6Suite. This module is in the network layer and responsible for checking the validity of IPv6 packets before finally forwarding them on. For this reason, all packets forwarded by IPv6 routers enter this module to be checked making it the optimum place to add the traffic codes.

It is important to be aware that since every node in the network has an IPv6Output module, we need to use some judgment before starting the detection function. That is, it should be confirmed that the current node is an edge router before starting the detection module on it.

2. Initialization module

This module is also placed on the edge routers of the network. It is mainly responsible for doing a series of initial operations when starting the tracing process. These initial operations include assigning and sending link signatures for each link in the network, signaling tracing intention, statically reconstructing the tracing information etc. This module is implemented as one of the applications of the IPv6Suite and faces two key issues.

1) Link signature assignment

Figure 3 illustrates the flow of link signature allocation. The blue arrows stand for the messages used to communicate with other modules.

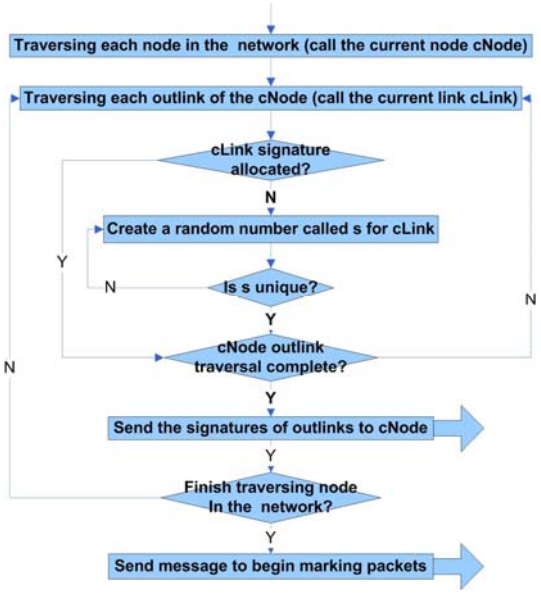

Figure 3. Flowchart of the link signature allocation algorithm

2) Reconstruction of static tracing information

In statically reconstructing the attack path each host in the network must be traversed to find the shortest path to the victim using the Dijistra algorithm. This allows for calculation of the tracing information. First the hops of this path are retrieved, then we can calculate the path signature of this path by XORing all link signatures in it. Because of the commutative law of the XOR operation, the result of XORing has nothing to do with the process. So the path signature calculated at this time should be equal to the value of the signature area of the packets which traversed the path. Coupled with the restriction of hop value, the two values are highly accurate in identifying the actual attack path. Experimental results show that the mis-tracing rate is less than 0.1%. Figure 4 illustrates the flow of the static path reconstruction process.

3. Information module

This module is on every router of the network. It is mainly responsible for saving the signatures of the current router's out links and the tracing intention of victim. When forwarding a packet, the router handles this packet according to this saved information. This module is also implemented as a application of IPv6Suite.

4. Sign module

This module is also on every router of the network and is mainly responsible for marking the packets to be traced. There is a key issue that needs discussing here. As already mentioned DDoS attacks are usually detected by the victim's traffic profile, specifically a burst followed by a smooth period. But in our actual network, there are many reasons besides DDoS attacks for traffic bursts, for example FTP and video frequency transactions.

Taking these situations into consideration, the sign module should firstly confirm whether or not the packet is part of normal traffic before marking it. According to the IPv6 specification, if the value of the traffic class area of IPv6 the headers is 0 this means that this packet is normal traffic [5]. Otherwise it belongs to some special traffic class.

As described for the detection module, the sign module is implemented by extending the IPv6 Output module.

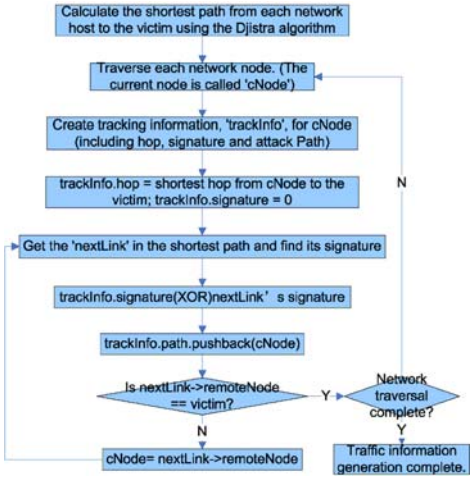

Figure 4. Flowchart of static tracking information generation

5. Reconstruction module

The reconstruction module is run on the edge routers of the network. A benefit of the static reconstruction algorithm is that no extra work need be done to this module to find from

the tracing information, the path whose signature and hops are equal to corresponding values in the attack packets.

We can deduce from the reconstruction process that the victim can reconstruct the entire attack path from just one attack packet. This seems efficient in theory, but in an actual environment, we can't arbitrarily treat hosts which send packets to the victim as attackers. This will lead to a high error rate.

To lower the rate of false positives, the number of times packets displaying a suspected of attack path arrive is taken into account. If the number of packets with the same path exceeds a threshold in a short time, we can then reasonably consider this path to be an attack path.

4.3. Simulation results

There were 15 IPv6 nodes in our simulation network, which included 6 IPv6 hosts and 6 IPv6 routers. Their topology is shown in figure 5.

We designate client0 as the victim and client3 as the attacker. 30 seconds into the simulation, the attacker initiates an ICMPv6-Flood attack with an interval of 0.1s. According to the shortest path first routing algorithm, the actual attack path is router3->router2->router1. The corresponding IPv6 addresses of these forwarding routers are b626:7:c88:84d4:21c1:1fd0:38c6:2c1a:6f3c-> edef:993c:d4e9:ce37:e895:c260:c3bc:56b4-> f853:3ae2:ccd:fc34:d8c9:c5ae:bdaf:436d. The main steps in our simulation process are.

1. The detection module on router0 detects the DDoS attack at the simulation time 30.000305s. Almost at the same time, it sends a message to the initialization module on router0 to start the tracing process.

2. The initialization module on router0 receives the start message at 30.000309s. This message starts a series of initialization operations. These operations are as follows.

- Assigning link signatures for each link in the network and send each router its out link's link signature.
- Send the tracing intention to each router.
- Statically reconstruct the tracing information.

Figure 6 shows the results of the static tracing information reconstruction.

3. The information module on each router receives the messages at the times shown in Table1.

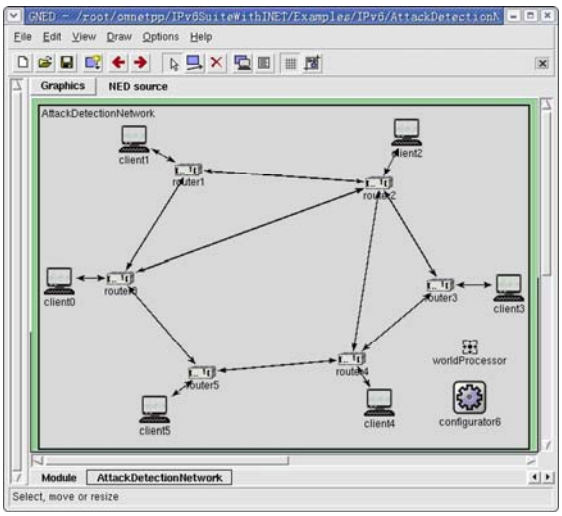

Figure 5. Topology of the Simulation Network

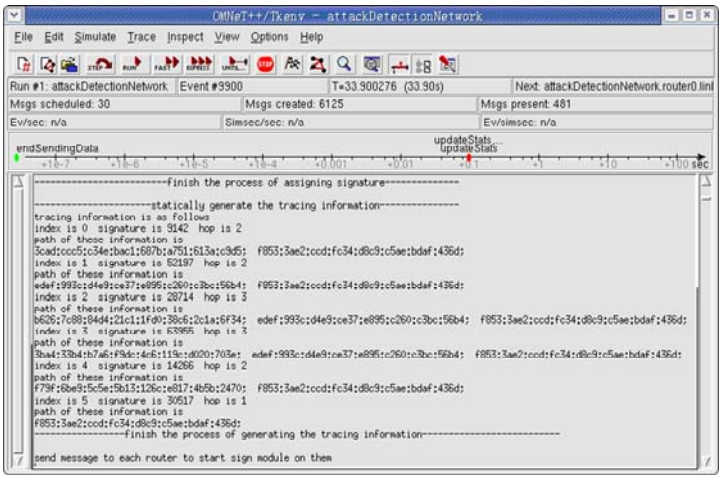

Figure 6. Tracing information generated by the initialization module

Table1. Times tracing information is received by each router

	Link Signature	Tracing Intention
router0	30.000311	30.000317
router1	30.000405	30.000437
router2	30.000369	30.000529
router3	30.000499	30.000594
router4	30.000597	30.000628
router5	30.000409	30.000441

4. The sign module on each router begins to mark the attack packets. The last router in this attack path sends the marked packets to the reconstruction module. The marking process on router0 is shown in figure 7.

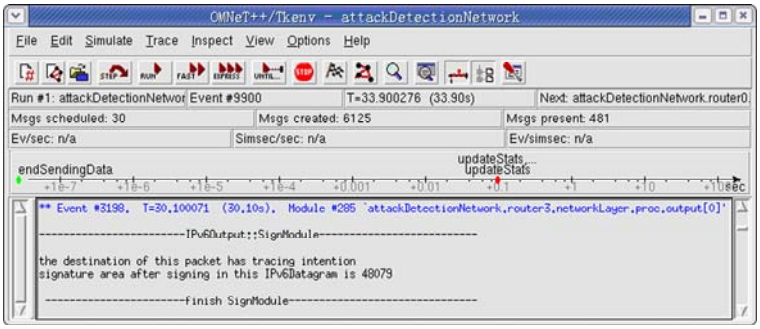

Figure 7. The Sign Module marks a forwarded packet

5. The reconstruction module on router0 identifies the attack path exactly at 30.200209s. This result is shown in figure 8.

We can observe from the simulation results that our tracing algorithm exactly reconstructed the attack path in 0.199904 seconds after the detection module detected the DDoS attack. We should recognize thought, that these results do not prove anything other than the veracity of our signature based DDoS attacker tracing algorithm. This is because the two thresholds applied in our simulation are not reliable in field application. For an actual network, more reliable thresholds are required to improve the algorithm's effectiveness.

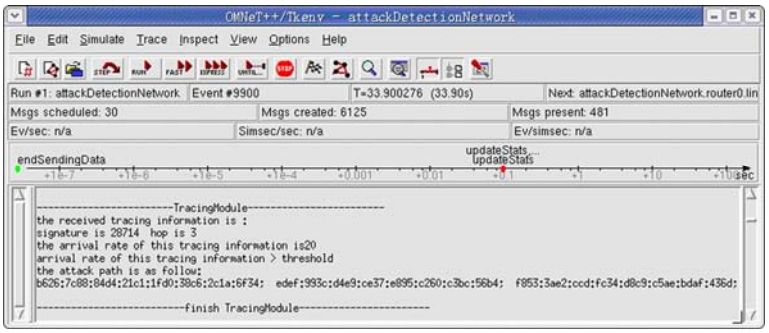

Figure 8. The tracing module reconstructs the attack path according to the received marked packets

5. Conclusion

In this paper we propose and show the veracity of a new signature based DDoS attacker tracing algorithm under IPv6. Although our simulation can only prove the veracity of this algorithm, it shows relative efficiency compared to other tracing algorithms in theory. In future, we will focus on the practicability of this algorithm and making it simpler and more effective in actual IPv6 networks.

References

[1] Song D. X. and Perrig A, Advanced and Authenticated Marking Schemes for IP Traceback, Proc. of INFOCOM, volume 2, 2001, pp. 878-86.

[2] Park K. and Lee H, On the Effectiveness of Probabilistic Packet Marking for IP Traceback under Denial of Service Attack, Proc. of IEEE INFOCOM, 2001, pp. 338-347.

[3] Adler M, Tradeoffs in probabilistic packet marking for IP traceback, Proc. 34th ACMSymp. Theory of Computing (STOC), 2002, pp. 407-418.

[4] Xinyu Yang, Ting Ma, Yi Shi, Typical DoS/DDoS Threats under IPv6, ICCGI.2007, March 2007 pp. 50-55.

[5] S. Deering and R. Hinden, Internet Protocol Version 6(IPv6) Specification, RFC2460, Internet Engineering Task Force, December 1998.

[6] Jelena Mirkovic and Peter Reiher, A taxonomy of DDoS attack and DDoS defense mechanisms, ACM SIGCOMM Computer Communication Review, April 2004, pp. 39-53.

[7] Christos Douligeris and Aikaterini Mitrokotsa, DDoS attacks and defense mechanisms: a classification, ISSPIT.2003, 14-17 Dec. 2003, pp. 190 -193.

[8] Y. Ahmet S_ekercio_glu, Simulation of IPv6 Networks with OMNeT++, ipv6-simulation.tex,v1.2.

[9] Johnny Lai, Eric Wu, Andr`as Varga, Y. Ahmet S¸ ekercio ˘ glu, Gregory K. Egan, A Simulation Suite for Accurate Modeling of IPv6 Protocols, 2nd International OMNeT++ Workshop January 2002, Berlin, Germany.

[10] Dawn Xiaodong Song, Perrig. A, Advanced and authenticated marking schemes for IP traceback, INFCOM.2001, Volume 2,22-26 April 2001 pp. 878-886 vol.2.

Authors

Ting Ma,author,Master of Xi'an Jiaotong University,is with the Electronic and Technology Department of Ningbo Dahongying University. happy_pepper@126.com.