2009 Ninth International Conference on Intelligent Systems Design and Applications

Design and Implementation of Distributed Intelligent Firewall based on IPv6

Qian Ma
College of Computer Science
Beijing University of Technology
Beijing, China
thewayma@gmail.com

Yingxu Lai
College of Computer Science
Beijing University of Technology
Beijing, China
laiyingxu@bjut.edu.cn

Guangzhi Jiang
College of Computer Science
Beijing University of Technology
Beijing, China
jiangguangzhi@gmail.com

***Abstract*—IPv6, as the alternative of IPv4, contains numerous features and improvements that make it attractive from a security perspective, but it is by no means the panacea for security. This paper presents the design and implementation of a distributed intelligent firewall system based on IPv6, which is able to secure the network layer and application layer of IPv6 networking. By the system, the typical attacks coexisting in both IPv4 and IPv6, the emerging IPv6 specific ones such as security threats related to ICMPv6, can be blocked by the rule set of network layer, similarly, with the rule set of application layer, any illegal or reactionary Web page content in HTML source codes can be totally prevented from sneaking into the Intranet. The Initiative Drift mechanism ensures the legitimacy and civilization of the Web environment within the whole IPv6 networking. Finally, we conduct the performance evaluation of the system and a decent result is gotten.**

Keywords-ipv6; icmpv6; firewall; network security

I. INTRODUCTION

Internet Protocol version 6 (IPv6) [1] is the next-generation Internet Layer protocol for the packet-switched Intranet and Internet. IPv6 developed as an alternative of IPv4 contains numerous features that make it attractive from perspective of the security. It is reliable and easy to set up, with automatic configuration. The huge address space with 128-bit makes it highly resistant to malicious scans and inhospitable to automated, scanning and self-propagating worms and hybrid threats [2]. Besides, the option fields in IPv6 header are handled as extension headers that give the flexibility to the IPv6 packet configuration. Some of those extension headers, such as AH or ESP headers [3], can provide confidentiality and integrity of network traffic.

Generally, IPv6 is more resistant to some security threats than the IPv4 protocol. But IPv6 is by no means the panacea for security, there are many security threats against IPv4 networks and some emerging threats [4-6] specific to IPv6, which both compromise the IPv6 networking. Indeed, some of the security threats are caused by some pitfalls and flaws of IPv6 related specification itself. A large number of IPv6 security holes, ever found so far, are mostly centered on flaws of Neighbor Discovery Protocol (NDP) [7], such as DoS, Man-In-Middle, etc. Furthermore, because the Internet Protocol Security (IPsec) protocol, as the mandatory component of IPv6, is implemented using the Authentication Header (AH) extension header or the Encapsulating Security Payload (ESP) extension header, the truly communication endpoint is only able to decrypt those encrypted messages and any network traffic encrypted by ESP are totally transparent to the firewall as the intermediate node. Generally, the technology of firewall applicable to IPv4 can not apply to the IPv6. So any adversary taking advantages of those security holes may easily sneak into IPv6 Intranet and compromise the whole networking without a powerful and innovative IPv6 firewall.

In this paper, we design and implement an innovative IPv6 distributed intelligent firewall system (DIFS) which is able to secure the whole IPv6 networking. The distributed system consists of three parts: the Administrator Center (AC), the Packet Filter Firewall (PFF), and the Web Page Analyzer (WPA). AC plays the core part within the system, it's responsible for the generation, maintenance and deployment of the policy. PFF exactly enforces the policy deployed by AC. WPA, as its name implies, is to analyze all the content of Web page which are either encrypted by ESP extension header or the typical unencrypted network packets. The detailed process will be explained at Section Ⅲ and Section Ⅳ.

The remainder of this paper proceeds as follows. In Section Ⅱ, we analyze IPv6 vulnerability. Section Ⅲ describes our comprehensive design of the prototype and highlights the functionality of the firewall system we implement. In Section Ⅳ, we elaborate the implementation of the system and conduct the evaluation of it. Finally, we conclude with the summary and future work in Section Ⅴ.

II. IPV6 VULNERABILITY

In order to design an innovative IPv6 firewall, it's necessary to analyze the security threats, especially the emerging IPv6 specific ones. This section is to analyze the IPv6 specific threats.

IPv6 specific attacks ever found are almost centered on the flaws of the Neighbor Discover (ND) protocol which is used to determine relationships between two of neighboring nodes, and designed to replaces ARP, ICMP router discovery [8], and ICMP Redirect message used in IPv4. Specifically, there are three ND protocol messages by which the packet of IPv6 attack is sent out to the wire, including Router Advertisement (RA, ICMPv6 type 134), Redirect (type 137) and Neighbor Advertisement (NA, type 136) which are to indicate MAC addresses, on-link network prefixes, on-link MTU information, redirection data, etc.

Supported by Scientific Research Common Program of Beijing Municipal Commission of Education (No: KM200810005030), and National Basic Research Program of China (973 Program) (No: 007CB311100).

978-0-7695-3872-3/09 $26.00 © 2009 IEEE
DOI 10.1109/ISDA.2009.151

Generally, all the attacks by counterfeiting router are commonly associated with RA or Redirect, so we call this kind of attack the counterfeiting router (CR) attack, while the attack by counterfeiting some single node is belonging to the counterfeiting node (CN) attack.

A. *Counterfeiting Router Attack*

The typical CR attacks include Bogus On-Link Prefix (BOLP), Bogus Address Configuration Prefix (BACP), Parameter Spoofing (PS) and PMTU Attack (PA), which all counterfeit a legitimate router to send the RA packet with the malicious parameters. For instance, BACP sends a false RA message and specifies an invalid subnet prefix to be used by a host for stateless address Auto-configuration [9][10].

B. *Counterfeiting Node Attack*

The typical CN attacks include Man in Middle (MIM) and Duplicated Address Detection (DAD) spoofing, which all send the illegal NA network packet to launch attacks. MIM just sends the illegal NA packets to the attacked hosts to become the man in middle and with the same link address as the attacked hosts, DAD spoofing can compromise the mechanism of IPv6 stateless address Auto-configuration.

Given all the analysis of the IPv6 specific attacks, it is obvious that the key to security is how to authenticate the senders of both RA and RA. The specific solutions will be detailed at Specification of Rule Set of Part B of Section Ⅳ.

III. DESIGN OF DISTRIBUTED INTELLIGENT FIREWALL SYSTEM BASED ON IPV6

A. *Architecture*

DIFS (shown in Figure 1) consists of three parts: the Administrator Center (AC), the Packet Filter Firewall (PFF) and the Web Page Analyzer (WPA). The system is maintained as the structure of a tree, namely every PFF can interconnect either with the IPv6 hosts directly, or with the sub packet filter firewall (Sub-PFF) embracing the different security demand and security level in some region. The security level of PFF is defined as Table 1.

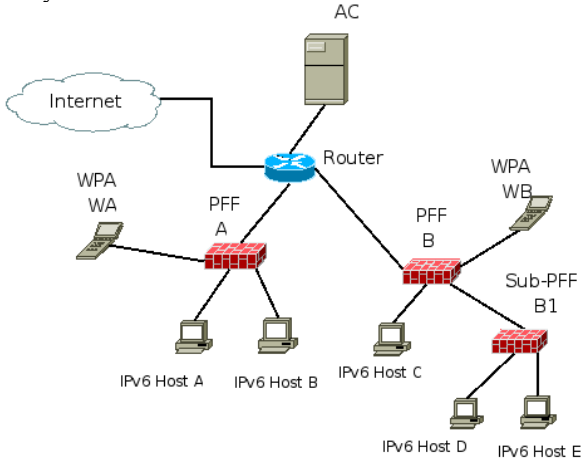


Figure 1. Architecture of the DIFS

TABLE I. SECURITY LEVEL OF PFF

Security Level	Default ACL	Other rule
Low	To accept all the traffic, unless prohibited by rule.	To enforce all the policy.
Medium	To prohibit some known dangerous ports	To enforce all the policy.
High	To deny all the traffic, unless accepted by rule.	To enforce all the policy.

1) *Administrator Center*

AC plays the core part within the system, it's responsible for the generation, maintenance and deployment of the policy. There are two kinds of policies called the packet filter firewall policy (PFFP) and the Web page analyzer policy (WPAP), as their name s imply, which are deployed to PFF and WPA, respectively. To ensure the validity and real-time of the policy, AC is able to deploy the policy periodically with the ever-changing networking.

2) *Packet Filter Firewall*

With different security level described in Table 1, PFF is able to embrace different security demand by using different default firewall rule. PFF exactly enforces the policy deployed by AC, including the typical access control list (ACL), the trustable multiple address rule list and the illegal URL rule list, by which PFF is able to block the typical existing network attacks and the emerging IPv6 specific ones, such as CR and CN threats described in Section Ⅱ, and prevent any illegal or reactionary Web content in HTML source code from into the Intranet, respectively. In addition to the periodical policy advertisement of AC, any PFF or Sub-PFF can also initiatively query and enforce the real-time policy.

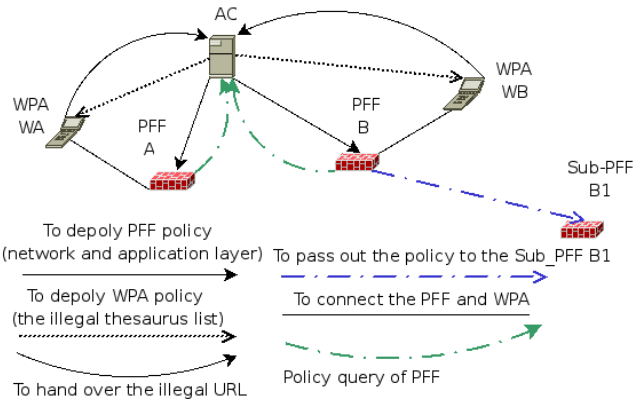
Basically, PFF performs the following tasks: A) To report its own security level to AC; B) To update the policy from AC; C) To enforce the policy, namely to put the ACLs and the illegal URL rule linked list into filter module of PFF; D) To query the policy initiatively; E) To filter the network packet one by one, given the rule sets of policy.

3) *Web Page Analyzer*

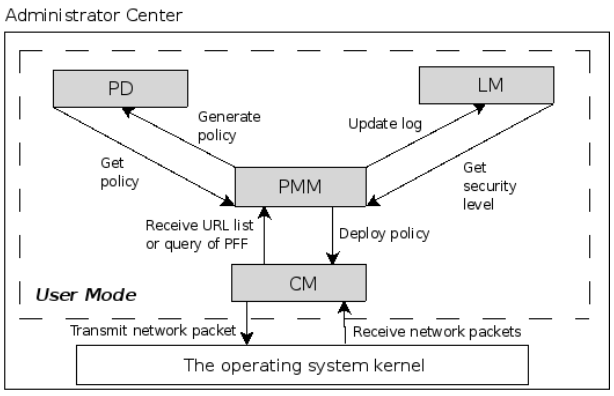
WPA is to analyze all the Web page content in HTML source code browsed by all the IPv6 hosts in some region (for the network traffic encrypted by ESP, after decryption, the truly IPv6 communication endpoint will hand over its browsing URL to WPA), then report the URL with illegal Web content to AC, which in turn will generate the policy dynamically, and then deploy the newly created policy to the correspondent PFF that will enforce the received URL rule list to block the illegal or reactionary Web content globally. Also, WPA will enforce WPA policy deployed by AC to control the analysis of Web content.

4) *Initiative Drift Mechanism*

Given the description above, we can draw conclusion that Initiative Drift Mechanism (IDM), characteristic by the global legal Web content guaranteed by any one of WPA which has ever found the illegal Web content, can bring us a



Firgure 2. Policy data flow chart of the system



Firgure 3. Module of AC

civilized Web environment within IPv6 networking.

By the typical method in intelligent techniques from AI domain, DIFS is an intelligent system capable of automatically indentifying the unknown illegal URL. Specifically, with the help of statistics and probability in AI, we dynamically maintain a global illegal thesaurus list which is controlled by AC's WPA policy, and the IDM enables every PFFs and sub-PFFs to enforce the global real-time illegal URL rule list and intercept any illegal Web content. For example, if the WPA WB (shown in Figure 2) catches a new illegal URL after matching between the global illegal thesaurus list and Web content, and then WB immediately send the result to the AC which in turn dynamically update the global illegal URL lists and deploy the newly created illegal URL rule list to all PFFs and Sub-PFFs. In short, the combination between IDM and self-learning against all new illegal URL brings us a civilized Web environment.

5) Data Flow of Policy

As the Figure 2, according to the real-time environment of IPv6 networking, AC must generate the policy dynamically and deploy the policy to the right place. To secure the networking all the time, AC deploys two kinds of policy to PFF and WPA periodically.

In addition to the periodical policy advertisement of AC, all the PFFs and sub-PFFs are able to query the policy initiatively.

Web Page Analyzer will enforce the policy from AC to control analysis of HTML Web data, and the Initiative Drift mechanism ensures the legitimacy and civilization of Web page content within the whole networking.

IV. IMPLEMENTATION AND TESTING

A. Administrator Center

Communication module (CM) is responsible for the network communication with PFF and WPA, including the deployment of policy, receipt of the policy query from PFF and receipt of URL list reported by WPA.

Log module (LM) is to keep track of the security level of PFF and any attack events.

Policy database (PD) is the medium of the specific rule set, including ACLs and illegal URL rule list, which are both stored in the XML files.

Policy maintenance module (PMM) is the interface to access and modification to PD, besides, as the most important one, PMM must be able to define the priority of ACLs.

B. Packet Filter Firewall

The firewall filter module of PFF is described as Figure 4. We conduct our research on Debian5.0 (kernel version 2.6.26-1-686) and the firewall system is built based on Netfilter [11] which is a framework that provides a set of hooks within the Linux kernel for intercepting and manipulating network packets. Besides, we write a character device [12] [13] in Linux kernel and finish the correspondent device driver in order to transmit the data between the Linux user mode and kernel mode. Thus, the policy of AC, including the typical firewall rules, the trustable multiple address rule list and the illegal URL rule list, can be totally injected into kernel, and then the kernel-level firewall codes can intercept and filter every network packet according to the linked lists of those rule sets.

1) Specification of Rule Set

ACL consists of different fields representing information relevant to the filtering. There are source IP, destination IP, source port, destination port, protocol and action. The rules are processed top down, and the first match is the only rule that applies.

The trustable multiple address rule list, periodically maintained by AC, represents the valid and trustable address of important servers, including the addresses of router and the DHCP server, etc. The firewall system only accept the Router Advertisement packet and IPv6 DHCP network packet whose source IP matches with this rule list. This rule list enables us to authenticate the sender of RA packet and just figure out the CR attack discussed at Part A of Section Ⅱ.

To avoid the CN threat explained at Part B of Section Ⅱ, we utilize the ip6_queue [14] Netfilter kernel module to verify the sender of IPv6 NA packet (shown in User Mode of Figure 4). Whenever the kernel intercepts NA packet, we just leverage ip6_queue module to get the socket buffer of that packet in User Mode. Then we extract the pair: source MAC and IP address of this packet and send the Neighbor Solicitation packet with the pair as its destination MAC and IP. Next, we just wait for another NA from the truly node and extract the source MAC of newly received packet. Finally, inject the comparison result of received MAC at two times into the kernel. The truly node will send the identical MAC in NA packet twice, while the hacker's different MAC means the first one stands for the counterfeited node, the second one stands for hacker's own one.

C. *Web Page Analyzer*

The algorithm of analysis of HTML Web data is the typical string match between those Web data and the global illegal thesaurus list which is updated by AC and all the programs in this part are programmed in Python. In order to get the URL which maybe encrypted by ESP, it needs the IPv6 host to use PF_PACKET socket to sniffer the browsed URL, then report the URL to WPA, which in turn can analyze Web page content given the received URL, finally report the URL with illegal Web content to AC.

D. *Testing*

We conduct the comprehensive evaluation of DIFS by SmartBits 6000c, which is the industry standard for network performance analysis. Because all the firewall rules are processed top-down, the performance of firewall system is basically inversely proportion to number of firewall rules theoretically and the number of firewall rule set of the network with medium size is basically less than 200, so number of rules should be set as little as possible while meeting the security requirement. In order to test the performance of firewall system processing different number of rule-sets, we just design 20, 40, 60, 80 and 100 firewall rules respectively which are all effective and optimized enough, we also make sure that the decision of every IPv6 network packet must be made by the last rule of ACLs. Then, according to those firewall rules with different numbers, we test the Throughput, Latency, Packet Loss and Back-to-Back of the system by SmartBits using 64 bytes, 128 bytes and 256 bytes frames. The performance curves of firewall system are shown from Figure 5 to Figure 8.

As shown above, the performance results are satisfactory. We can make use of more than 50% bandwidth at the worst case, namely the tiny network packet with 64 bytes and 100 firewall rules to be enforced and the result is up to 100% as long as the packet is 256 bytes. Given this result, we can also fully use the bandwidth when the more rules is going to be enforced. In Figure 6, the biggest latency is no more than 290 μs when big packet with 256 bytes to be sent and 100 rules to be enforced, but the result also looks sound while taking into consideration that DIFS is a firewall system rather than router. Furthermore, in terms of the tiny packet with 64 byte executed by firewall frequently, the latency is no more than 110 μs. At the speed of 100% bandwidth, the packets will be lost by 50% at worst when 64 bytes and 100 rules, after all our PFF is just built on the common PC with two 100Mbps NIC. It's no doubt that the better result will be gained only if we improve the hardware.

V. CONCLUSION

In this paper, we have presented the design and implementation of DIFS, a Distribute Intelligent Firewall System based on IPv6 that can secure the whole IPv6 networking real-time. The novelties of DIFS are the whole distributed architecture suitable for the emerging IPv6 networking, self-learning against the new illegal URL and the Initiative Drift Mechanism which is able to ensure the legitimacy and civilization of the Web environment within the whole IPv6 networking. Our prototype implementation and experimental results have shown us a decent firewall system which can be deployed to the practical networking.

REFERENCES

[1] S. Deering, R. Hinden, "Internet Protocol, Version 6 Specification", RFC 2460, Dec. 1998.

[2] M.H. Warfield, "Security Implications of IPv6", Internet Security System, 2003.

[3] M.R. Sabir, M.A.Fahiem and M.S. Mian, "An Overview of IPv4 to IPv6 Transition and Security Issues", CMC '09. WRI International Conference, 2009(3):636-639.

[4] B.H.Jung, J.D.Lim, Y.H.Kim and K.Y.Kim, "An Analysis of Security Threats and Network Attacks in IPv6", Electronics and Telecommunications Trends,2007,22(1): 37-50.

[5] D. Zagar, K. Grgic and S. Rimac-Drlje, "Security aspects in IPv6 networks",Computers and Electrical Engineering. 2007(33):425–437.

[6] C.E. Caicedo, J.B.D. Joshi and S.R. Tuladhar, "IPv6 Security Challenges", Computer, 2009,42(2):36-42.

[7] T. Narten, "Neighbor Discovery for IP version 6 (IPv6)", RFC 2461 on IETF, Dec. 1998.

[8] S. Deering, "ICMP Router Discovery Messages," RFC 1256, Sept. 1991.

[9] S.Thomson, T. Narten, "IPv6 Stateless Address Aut-oconfiguration", RFC 2462 (Draft Standard), Dec. 1998.

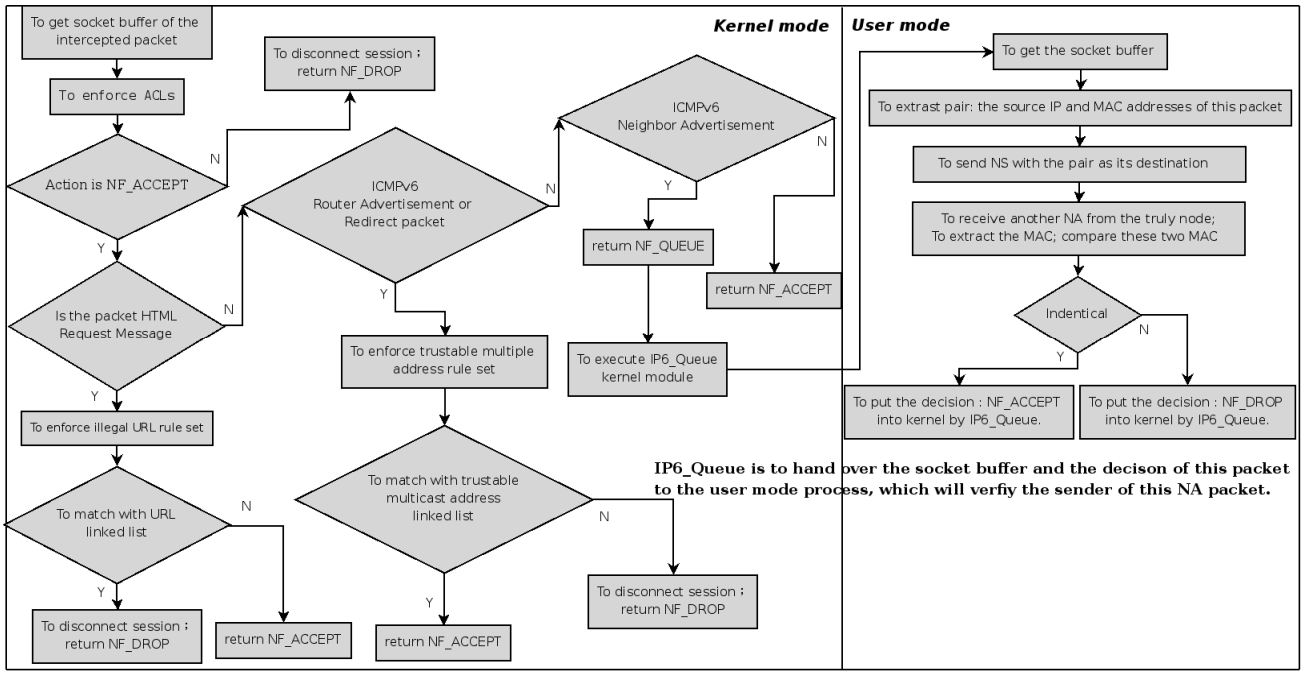
[10] T. Narten, R. Draves, "Privacy extensions for stateless address Autoconfiguration in IPv6", RFC 3041, IETF, January 2001.

[11] Netfilter: A set of hooks inside the Linux kernel that allows kernel modules to register callback functions with the network stack. http://www.netfilter.org.

[12] P.B. Daniel, C. Marco, Understanding the Linux Kernel, 3rd Edition,O'Reilly, Page 517-553,November 2005.

[13] J. Corber, G. Kroah-Hartman, A. Rubini, Linux Device Drivers, 3rd Edition,O'Reilly, Page 33-41, February 2005.

[14] IP6_Queue: A userspace library providing an API to pa- ckets that have been queued by the kernel packet filter. http://www.netfilter.org/projects/libnetfilter_queue/index.html.



Firgure 4. Flow chart of kernel-level firewall

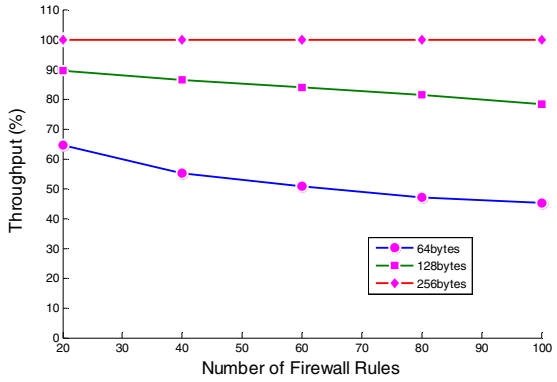


Figure 5. Comparison curve of Throughput

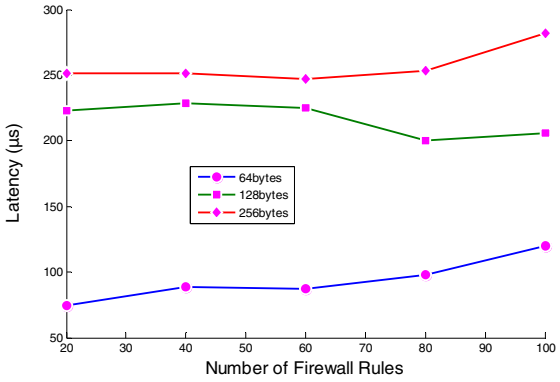


Figure 6. Comparison curve of Latency

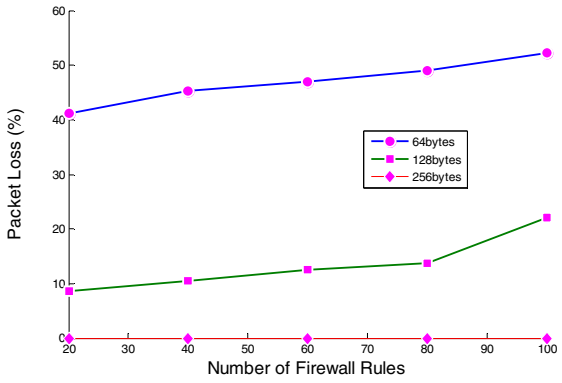


Figure 7. Comparison curve of Packet Loss

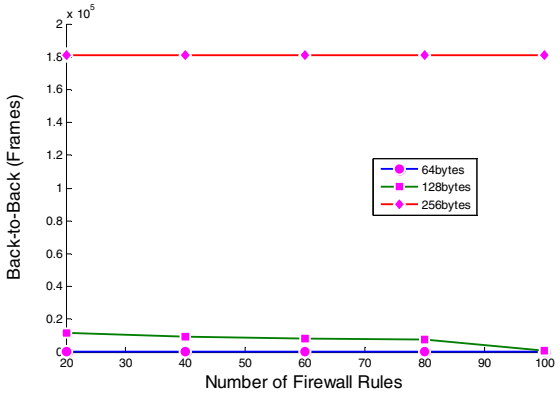


Figure 8. Comparison curve of Back-to-Back