I.J. Wireless and Microwave Technologies, **2011, 2, 33-40**
Published Online April 2011 in MECS (http://www.mecs-press.net)
DOI: 10.5815/ijwmt.2011.02.05

Available online at http://www.mecs-press.net/ijwmt

A Novel Approach to Simulate DDoS Attack

Qing LI, LeJun CHI, ZhaoXin ZHANG

School of Computer Science and Technology, Harbin Institute of Technology at WeiHai, WeiHai, China

Abstract

Due to the rapid spread of botnets, distributed denial of service attacks have become more and more frequent and fatal. In order to detect and defend DDoS attack, many researches have been done, such as using NS2 to simulate such attacks. However, the major bottlenecks in large-scale network simulation are the memory usage and simulation time. In this paper, we present a novel approach to simulate DDoS attack, called Analytical Packet Forwarding Model (APFM), which simplifies the procedure of packet forwarding based on computation. The experimental results show that our model ensures the authenticity of the overall simulation procedure and simulation results, compared to NS2. The most significant contribution of APFM is that it significantly reduced the memory usage and simulation time.

Index Terms: Distributed denial of service attack; simualtion; packet forwarding

1. Introduction

Researches of DDoS attacks started since its birth. These researches provide methods to detect DDoS attacks used by firewall or network-based intrusion detection systems (NIDS), so simulating DDoS attack is essentially needed. Developing the specific software [1] or using the existing network simulation software, such as OPNET [2], NS2 [3], can help us to build a virtual environment. However, most existing network simulation software have severe limitations, especially the memory and CPU time requirements. Simulated networks of just a few thousand network elements and a few thousand flows will quickly exhaust the computing resources in any reasonably sized computer workstation [4]. George and Mostafa presented that assuming we want to simulate 100 seconds activity on the Internet we need more than a year of CPU time nearly three hundred terabytes of main memory(not include memory needed for simulated routing tables at each node) [5]. So, in order to simulate large-scale networks behavior like DDoS attack of Internet, one reasonable solution is reducing the memory usage and simulation time.

In this paper, we present a novel method to simulate DDoS attack, referred to as the Analytical Packet Forwarding Method (APFM), which simplifies the procedure of packet forwarding based on computation. We

* Corresponding author.
E-mail address: lq1986627@163.com, qdclj@163.com, heart@hit.edu.cn

take advantage of mathematical model to describe the procedure of packet forwarding. Our model captures the characteristics of DDoS attack. To evaluate our method in practice, we compare it to the network simulator—NS2. Experimental results show that our approach can effectively simulate DDoS attack.

The remainder of this paper is structured as follows. Section II describes the simplex link that is a major compound object in NS2, and the course of packet forwarding between nodes in NS2. In Section III, we present the APEM. In Section IV, we compare it to the NS2. We conclude this paper in Section V with a brief summary and an outline of our future work.

2. Overviw of NS2's Simplex Link Model

NS2 is the second version of a network simulator tool developed by the Virtual Inter Network Testbed (VINT) project [6]. It is a discrete event-driven network simulator, which is popular with the networking research community.

Fig.1 shows the simplex link in NS2. It is formed by three objects; they are Queues, DelayLink and TTLChecker.

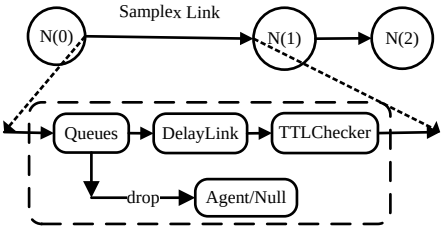

Figure 1. Simplex link in NS2[7]

There is one thing to note is that an output queue of a node is actually implemented as a part of simplex link object in NS2. As showed in Fig.1, the process of a packet is transmitted from node N (0) to node N (1) describes as follow: when a packet was delivered to the simplex link from node N (0). Firstly, if the inputs queue of the Queues object has enough room for this packet, the packet will enqueue and wait for handling. If not, the packet will be dropped, and the packet will be delivered to the Agent/NULL object and is freed there. When it is turn to this packet, this packet will dequeued from the input queue and is passed to the DelayLink object that simulates the link delay. Then, the DelayLink object will generate a new event based on the link delay and insert the event into an event queue in Queues object. When the right event is triggered the packet will be passed to the TTLChecker object. Finally, the TTLChecker object calculates Time to live parameters for each packet received and updates the TTL field of the packet. If the packet's TTL value is 0, it will be dropped at a queue is sent to the Agent/NULL object and is freed there. If not, the packet will be received by node N (1).

3. Analytical Packet Forwarding Modle (APFM)

In Section II we can see, on the one hand, the procedure of packet forwarding in NS2is not continuous, but intermittently. The packet need constantly to enter and out from different queues in the Queues object, wait for next operation and is triggered until the right time. Since NS2 is an event-driven network simulator. It causes too much computational overhead.

There is one thing to state is that the background traffic is not included in APFM and the DDoS attack simulation is based on UDP protocol. In order to reduce the memory usage and simulation time, we employ a simplified and abstract method called mathematical modeling to describe the packet forwarding procedure based on computation. By analysis the DDoS attack, we build a corresponding mathematical model.

A. The Procedure of Packet Forarding in APFM

NS2 is a packet-level network simulator. When we simulate the behavior of networks such as DDoS attack using NS2, by describing the procedure of packets forwarding from sources node to destinations node one by one, including the time of packets arrived each node, the delay of packets at node and the total number of packets have forwarded and the total number of packets are dropped at the simplex links between nodes, and so on.

Following is the procedure of packet forwarding in APFM, which describe a packet forward process from node N(0) to node N(1) as showed in Figure 1. In the remain of this paper we assume the simplex link from N(0) to N(1) as a object of N(0). Similarly, the simplex link from N(1) to N(0) as a object of N(1). It is different from node in NS2, but is consistent with the route in real network. Firstly, we need to define some parameters. Assuming Size(p) is the size of packet p, t(0) is the time packet p delivered to the simple link at node N(0), Length(t(0)) is the length of buffered queue of N(1) at time t(0), the bandwidth between N(0) and N(1) is B(0, 1)(Bytes/s), the communication delay is D(0,1)(s); the bandwidth between N(1) and N(2) is B(1,2)(Bytes/s), the communication delay is D(1, 2)(s); MAX_LENGTH is the max size of queues. According to these parameters, we can see:

- Use Equation (1), we can get *t(1)*, the time which packet *p* arrived N(1).

$$t(1) = t(0) + D(0, 1) \tag{1}$$

- Assume the queue length of N(1) at *t(1)* is *Length(t(1))*. Use Equation (2), we can judge the packet *p* whether dropped at N(1).

$$\begin{cases} Length(t(1)) + Size(p) > MAX\_LENGTH, dropped \\ Length(t(1)) + Size(p) \le MAX\_LENGTH, notdropped \end{cases} \tag{2}$$

- If packet *p* not dropped at N(1), first we should update the queue length of N(1), as follows:

$$\begin{aligned} &Length(t(1)) \\ &= sign(Length(t(0)) - B(1,2)(t(1) - t(0))) + Size(p) \end{aligned} \tag{3}$$

And

$$sign(x) = \begin{cases} 0, x < 0 \\ x, x \ge 0 \end{cases} \tag{4}$$

- Assume the delay which packet *p* at N(1) is *delay(p,1,2)*. We ignore the sending delay and processing delay at N(1).

$$delay(p, 1, 2) = Length(t(1))/B(1, 2) \tag{5}$$

So, packet *p* arrived N(2) at time *t(2)*,

$$t(2) = t(1) + delay(p, 1, 2) \tag{6}$$

B. Implement of APFM

The essence of APEM is employing direct calculation get the details of packets forwarding at each node from the sources to destinations. Since the sources (attack nodes), destinations (attacked nodes), packet size and attacking rate are known in DDoS attack simulation, so we can use those (1)-(6) to describe the activities of DDoS attack simulation.

In the implement of APFM, we use a priority queue to replace the work of scheduler in NS2. The process of APFM as showed in Fig.2.

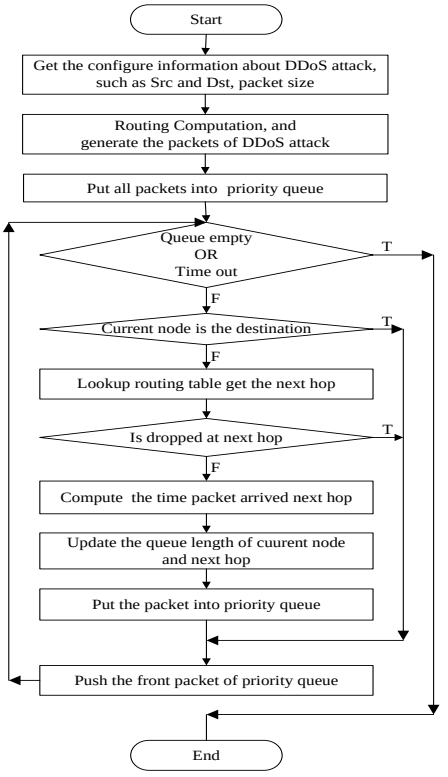

Figure 2. The Flow Chart of APFM

4. Experimental Results

To evaluate the authenticity and performance of APFM, we compare it to NS2. In this section, we describe details of the experiments and analyze the experimental results.

We ran our experiments on a Dawning Server with an Intel Xeon5500 processor 2.8GHz CPU, 4GB RAM, running Red Hat AS4 Linux operating system. Packet size is 530KB. Each experiment simulates 9 seconds DDoS attack with different attack node numbers and attacked node numbers, topology size and packets number. The details are showed in TABLE Ⅰ.

TABLE I. EXPERIMENT DATA

Topology Size	*Attack Node Number*	*Attacked Node Number*	*Packets Number*
8377	150	2	27000
9442	200	2	36000
9603	100	2	18000
13950	200	3	54000

C. Validation of APFM

In order to validate the credibility of DDoS attack simulation which employ APFM to simulate, we compare the time that packets forwarded at each node from source to destination, and the total number of packets have forwarded and dropped of nodes. The experiments results are showed in Fig.3, Fig.4 and Fig.5.

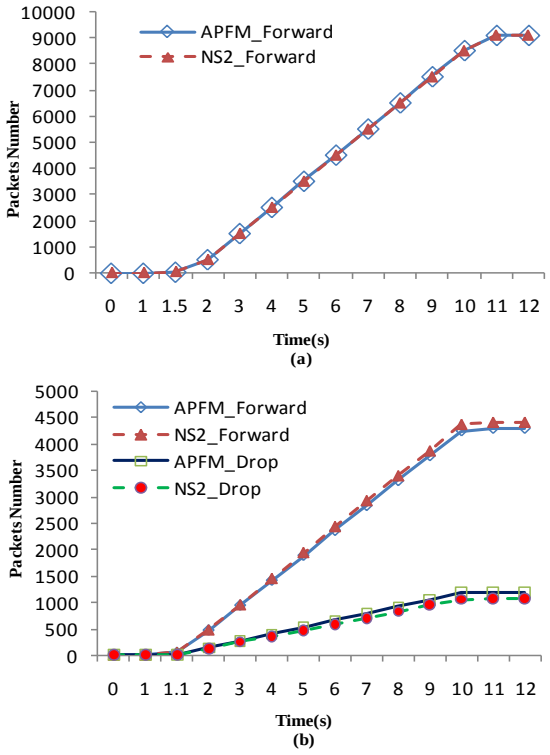

Figure 3. Packets Number of Node Forwarded and Dropped

Fig.3 gives the details of two nodes which are picked up randomly from topology during the overall simulation procedure. Fig.3-(b) shows that there are some packets are dropped. From Fig.3, we can see the results of APFM are very consentaneous to the results of NS2.

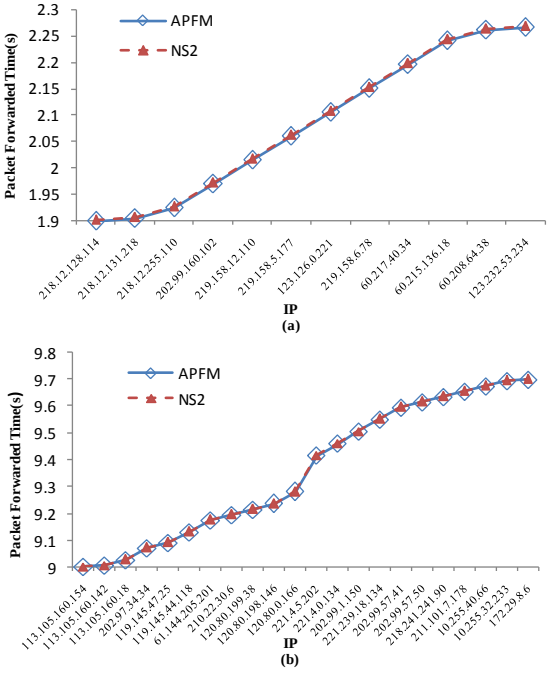

Figure 4. Packet Forwarded Time

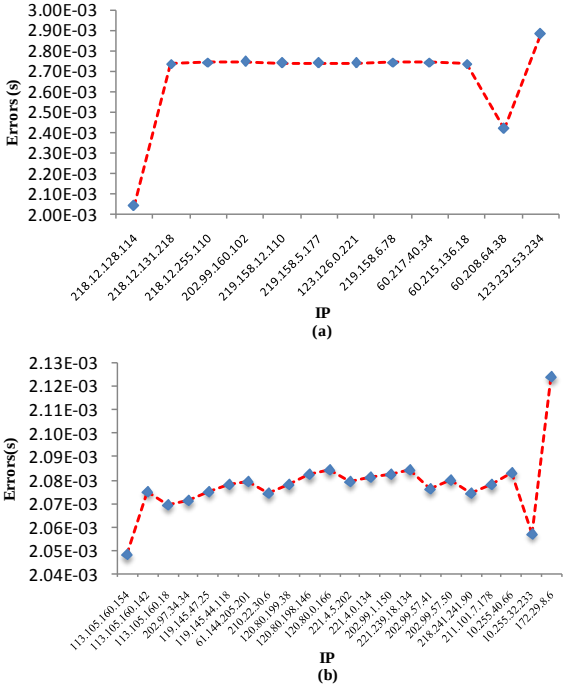

Figure 5. Errors of Packets Forwarded Time

Fig.4 shows the forwarded time of two packets at each node from source to destination. The errors between APFM and NS2 about the packet forwarded time at each node which reported in Fig.4 are showed in Fig.5. Form Fig.4 and Fig.5, we can see the errors vary from 0.002025 seconds to 0.0029 seconds. It illustrates the simulation result of APFM has highly credibility.

D. Simulation Performance of APFM

The performance of a simulation method can be indicated mainly by two parameters: memory usage and simulation time. In this subsection, we describe the experimental results of these parameters with different topology size and packet number reported in TABLE I.

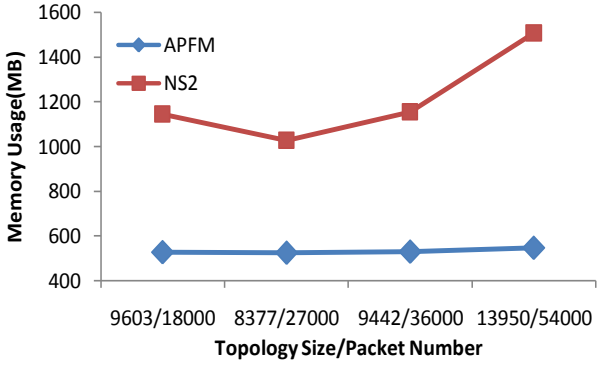

Figure 6. Memory Usages versus Topology Size/Packet Number

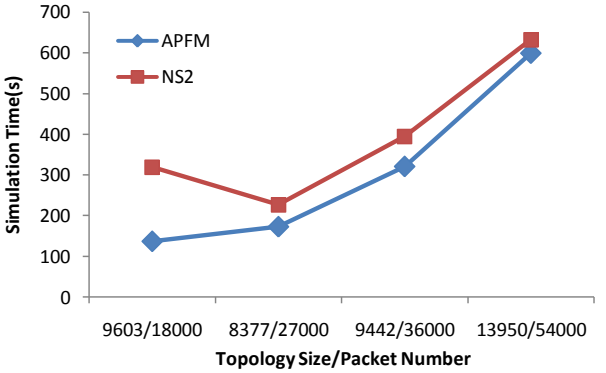

Figure 7. Simulation Time versus Topology Size/Packet Number

Fig.6 shows the memory usage with different topology size and number of packets by using APFM and NS2 to simulate DDoS attack. From Fig.6, we can see the memory usage of APFM did have great changes with different topology size and packet number, because we ignore the routing table of nodes which aren't on the paths of DDoS attack in APFM. On the contrary, as well as the general routing table, the simulator also creates some classifiers for each link according to the routing table which also require a large amount of memory. The result shows that the memory usage of APFM is less 55.6% on average than NS2.

Fig.7 shows the simulation time with different topology size and packets number. The graph shows that the simulation time of APFM is about less 26.3% than NS2. The reason is the routing computation of a part of nodes we ignore in APFM consummate some simulation time.

5. Conclusions and Future Work

In this paper we focus on how to reduce the memory usage and simulation time in DDoS attack simulation. We present the APFM to simulate DDoS attack. The simulation results show that APFM has very highly credibility, through there are certain errors compared to NS2. Based on the experimental results, APFM uses about 55.6% less memory and achieves a reduction in simulation time of about 26.3% compared to NS2.

In the future, we would take into account the background traffic so that it can simulate the cases which more approximate to the realistic DDoS attack.

Acknowledgment

This work is sponsored by the National High Technology Research and Development Program of China (863 Program) (2007AA010503). The authors would like to thank the anonymous reviewers for their helpful comments for improving this paper.

References

[1] I. Kotenko, A. Ulanov, "The Software Environment for Multi-agent Simulation of Defense Mechanisms against DDoS Attacks", Proceedings of CIMCA 2005 and International Conference on Intelligent Agents, Web Technologies and Internet, 2005, pp. 283-288.

[2] M. Q. Zhang, J. Xie, M. Zhang, X. L. Zhang, "Modeling and simulation of DDos attacks using OPNET modeler", Xitong Fangzhen Xuebao / Journal of System Simulation, v 20, n 10, May 20, 2008, p 2736-2739 Language: Chinese.

[3] H. W. Lee, T.Y. Kwon, H. J. Kim, "NS-2 based IP traceback simulation against reflector based DDoS attack", Proceedings of Artificial Intelligence and Simulation. 2005, pp. 90-99

[4] S. Floyd, V. Paxton. Difficulties in Simulating the Internet. IEEE/ACM Transactions on Networking (TON). 2001,9(4):392-403.

[5] G. F. Riley, M. H. Ammar. Simulating Large Networks How Big is Big Enough?[C]. In Proceedings of the First International Conference on Grand Challenges for Modeling and Simulation, San Antonio, TX, 2002:28-31.

[6] S. Bajaj, L. Breslau, D. Estrin, et al. Improving simulation for network research. Technical Report 99-702b, USC Computer Science Department, 1999.

[7] The Network Simulator ns-2: ns by Example; http://nile.wpi.edu/NS/.