AN INTELLIGENT METHOD FOR REAL-TIME DETECTION OF DDOS ATTACK BASED ON FUZZY LOGIC[1]

Wang Jiangtao Yang Geng*

(*College of Computer*, *Nanjing University of Posts and Telecommunications*, *Nanjing 210003*, *China*)

*(*Research Institute of Computer Technology*, *Nanjing University of Posts and Telecommunications*,
Nanjing 210003, *China*)

Abstract The paper puts forward a variance-time plots method based on slide-window mechanism to calculate the Hurst parameter to detect Distribute Denial of Service (DDoS) attack in real time. Based on fuzzy logic technology that can adjust itself dynamically under the fuzzy rules, an intelligent DDoS judgment mechanism is designed. This new method calculates the Hurst parameter quickly and detects DDoS attack in real time. Through comparing the detecting technologies based on statistics and feature-packet respectively under different experiments, it is found that the new method can identify the change of the Hurst parameter resulting from DDoS attack traffic with different intensities, and intelligently judge DDoS attack self-adaptively in real time.

Key words Abnormal traffic; Distribute Denial of Service (DDoS); Real-time detection; Intelligent control; Fuzzy logic

CLC index TP393

DOI 10.1007/s11767-007-0056-6

I. Introduction

When modeling bursting traffic in high speed network in communication field, the network arrival process is often assumed as Poison arrival. However, Bellmore's survey on LAN's packet traffic[1,2], as well as many research institute's analysis on internet traffic on Wide Area Network (WAN)[3–5], show that the process of bursting traffic arrival is much more coincident with exact or asymptotic self-similarity model than Poison arrival. The Hurst parameter is an important parameter used to describe the bursting feature of self-similarity network traffic. Lots of study shows that DDoS attack can exert remarkable influence on the self-similarity of network traffic; therefore, DDoS attack can be effectively detected by the change of the Hurst parameter[6].

Solutions for obtaining the Hurst parameter include Variance-Time Plots (VTP)[7], R/S analysis[1], Whittle method[8], wavelet transform method[9], *etc.* Among them, VTP's calculation and implementation is so complex and time consuming that it can not be used in real-time calculation. R/S analysis is an early method with high complexity and low precision. The Whittle method is quite precise and has confidence interval, but its complexity is the highest among these methods. Although wavelet transform method obtains the precise Hurst parameter, its calculation and implementation is as complicated as VTP, so that it cannot be used in real-time calculation as well.

Based on the VTP method, we introduce slide-window mechanism, which can greatly reduce the time of data sampling and computation, to increase the speed of calculate the Hurst parameter. Traditional DDoS judgment mainly depends on the known Hurst parameter and experience; it lacks self-adaptability and has great subjectivity. DDoS attack is a process that changes dynamically and frequently. We introduce fuzzy logic method to implement DDoS judgment; it improves the self-adaptability and the agility of DDoS judgment.

II. Definition of Traffic's Self-similarity

The definition of self-similarity[2] is described as

[1] Manuscript received date: March 19, 2007; revised date: June 28, 2007.

Supported by the Six Heights of Talent in Jiangsu Province (No.06-E-044).

Communication author: Wang Jiangtao, born in 1978, male, Ph.D. candidate. College of Computer, Nanjing University of Posts and Telecommunications, Nanjing 210003, China.

Email: wjtmas@163.com.

follows: Let $X=\{X(t), t=0,1,2,3,\cdots\}$ be a wide-sense stationary process with constant mean μ, finite variance σ^2, and autocorrelation function $r(k)$ that depends only on $k,(k=1,2,3,\cdots)$.

$$\left.\begin{aligned}\mu &= e[X(t)]\\ \sigma^2 &= E[(X(t)-\mu)]^2\\ r(k) &= E[(X(t)-\mu)(X(t+k)-\mu)]/\sigma^2\end{aligned}\right\}\qquad(1)$$

Let $X^{(m)}=\{X^{(m)}(t), t=0,1,2,3,\cdots\}$ denote the aggregate process of X at aggregation level m $(m=1,2,3,\cdots)$. That is, for each m, $X^{(m)}$ is given by

$$X_k^{(m)}=(X_{km-m+1}+\cdots+X_{km})/m,\quad k=1,2,\cdots;\ m=1,2,\cdots\qquad(2)$$

For each m, $X^{(m)}(t)$ is defined as a covariance stationary stochastic process and $r^{(m)}(k)$ is the autocorrelation function of $X^{(m)}$.

$$r^{(m)}(k)=r(k)\sim k^{-\beta},\ 0<\beta<1\qquad(3)$$

Then $X(t)$ is called exactly the second-order self similar with self-similarity parameter H=1$-\beta/2$. Here "~" means equivalent. While if X has an autocorrelation function of the form for all m:

$$r^{(m)}(k)\sim r(k),\ m\to\infty\qquad(4)$$

the $X(t)$ is called asymptotically the second-order self-similar with self-similarity parameter $H=1-\beta/2$.

III. Variance-time Plots Method

VTP cannot be used in the real-time detection of the Hurst parameter because its computational time is too long. In this section, we analyze VTP's performance mainly from the aspect of algorithm to find out the key problems which influence its computational time.

1. Descriptions of variance-time plots

Variance-time plots[7] are obtained by plotting $\ln(\mathrm{Var}(X^{(m)}))$ against $\ln(m)$ and by fitting a simple least square line through the resulting points in the plane while ignoring small values for m. The procedure can be formulated in the following:

(1) Divide the original time series $X=\{X(t), t=0,1,2,\cdots\}$ into blocks of size m and obtain the average within each block. Take the sample variance of $X_k^{(m)}, k=1,2,\cdots$, within each block. This sample variance is an calculate of $\mathrm{Var}(X^{(m)})$.

(2) Obtain H by the following sub-steps:

(a) For a given m, divide the data $X_1,\cdots,X_N$ into N/m blocks of size m, calculate $X_k^{(m)}$ for k=1, 2,$\cdots$,N/m, and its sample variance by

$$\mathrm{Var}(X^{(m)})=\frac{1}{N/m}\sum_{k=1}^{N/m}(X_k^{(m)})^2-\Big(\frac{1}{N/m}\sum_{k=1}^{N/m}X_k^{(m)}\Big)^2\qquad(5)$$

(b) Repeat (a) for different values of m. Plot the logarithm of the sample variance versus $\ln(m)$.

2. Computational cost of calculation of the VTP

There are two processes that cost much time in the calculation of the Hurst parameter:

(1) Every data is conglomerated to form new sequences and the average value of the new sequences is calculated. The computational cost of this procedure is S_1.

(2) Calculating the variance of the new sequences. The computational cost of this procedure is S_2.

Suppose N data in sequence X need to be analyzed, there are mainly two types of operation in Procedure (1), namely, addition and division. To each conglomerate degree $m=2,3,\cdots,\mathrm{INT}(N/2)$ the computational costs of addition and division are:

"Σ": (m–1) INT(N/2); "/": INT(N/m)

INT(x) denotes the function to get the maximum integer that is equal to or less than x.

So the total cost of Procedure (1) is

$$S_1=2\mathrm{INT}\left(\frac{N}{2}\right)+\cdots+\frac{N}{2}\mathrm{INT}\left(\frac{N}{N/2}\right)=\sum_{m=2}^{N/2}k\mathrm{INT}\left(\frac{N}{k}\right)\qquad(6)$$

The computational costs of the Procedure (2) are

(a) Computing all elements' quadratic sum and then the average of the sums;

(b) Computing the average of the sum of all elements and then the square of the average;

(c) Computing the difference of the two results.

So the total computational cost of Procedure (2) is

$$\begin{aligned} S_2 &= (3N+1)+\left\{3\mathrm{INT}\left(\frac{N}{2}\right)+1\right\}+\left\{3\mathrm{INT}\left(\frac{N}{3}\right)+1\right\} \\ &\quad +\cdots+\left\{3\mathrm{INT}\left(\frac{N}{N/2}\right)+1\right\} \\ &= 3\sum_{k=1}^{N/2}\mathrm{INT}\left(\frac{N}{k}\right)+\frac{N}{2} \qquad (7) \end{aligned}$$

We can find from Eq.(6) and Eq.(7) that it is more complex to calculate S_1 than S_2. So S_1 is a bottle-neck of VTP. In order to improve the performance of VTP, we should speed up S_1's computation to make VTP applicable in the real-time detection.

3. Calculate the Hurst parameter in Real-time VTP (RVTP)

We put forward a new approach which can reduce the computational time remarkably[7,10]. It calculates the Hurst parameter with RVTP.

Fig.1 shows how to calculate the Hurst parameter in network traffic. In Fig.1(a), l represents the original number of data package when calculate the Hurst parameter for the first time. The method which calculate the Hurst parameter in real time based on slide-window means to put the l original packets in the slide-window of certain size (Here we suppose the window size is $h, l = 10h$), and the slide-window can be resized dynamically according to demand. When detecting the Hurst parameter, we only need to abandon a certain amount of data from the head of the sequence and add data of the same number of data package to the tail, and then calculate the Hurst parameter once again. For instance, in Fig.1(b), we abandon the data of the first window and append a window to the tail to keep the data length unchanged. Thus every time we calculate the Hurst parameter, we only need to calculate the added data whose number of data package is h. Fig.1(c) is similar to Fig.1(b).

The computational cost of S_1 is decreased greatly using the RVTP method because we need only to calculate S_1 of the renewed data in instead of calculating all of the original data. So by choosing proper l and h and adjust the relationship between the renewed data and the original data, we can detect DDoS attack in real time.

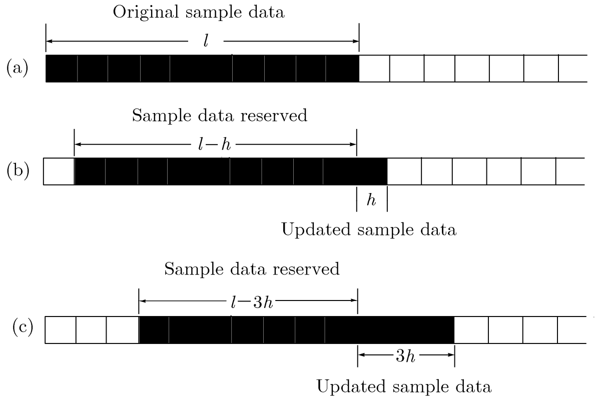

Fig.1 Calculating the Hurst parameter in real time

IV. Experiments and Analysis

In order to check the performance of the method presented in the paper, we apply the 1999 DARPA Offline Intrusion Detection Evaluation to carry out simulation experiments[11]. The hardware environment is: Athlon XP2500+CPU, 512MB memory.

In order to test the performance of RVTP in the detection of DDoS attack in general, we add DDoS attack traffic with different intensity and observe the attack traffic's influence to detect DDoS attack quickly. According to RVTP, we set the conglomerate degree m to be 1, 2, 3,$\cdots$, 9, 10, 20, 30, $\cdots$, 100, respectively, and then we draw the figure of $\ln(\mathrm{Var}(X^{(m)})) - \ln(m)$ to calculate the Hurst parameter[12].

1. Testing normal traffic sequence in real time

After choosing four groups of normal traffic sequences, the Hurst parameter and computational time of each sequence are got in the experiment. In Fig.2, we set the number of data package l=20000 (Its sampling time equals 10s). We draw curve "b" by abandoning the data package whose number is h (l=10h) of curve "a" and appending the same number to the tail of "a". Similarly, we draw curves "c" and "d" by replacing the data whose number of data package is h.

From Fig.2, we find the change patterns of these curves' slopes are almost the same. Therefore, as for the network traffic is produced when the network works normally, the Hurst parameter is basically stable and within the range of the normal self-similar model.

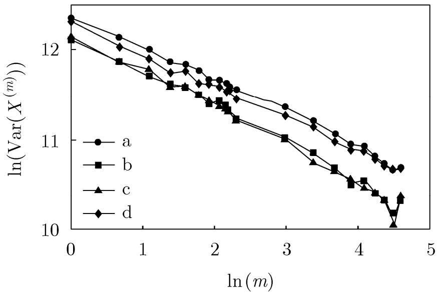

Fig.2 A plot of the $\ln(\mathrm{Var}(X^{(m)})) \sim \ln(m)$ for normal traffic

Tab.1 shows the Hurst parameter of each sequence and their computational time. It indicates that, when we calculate the Hurst parameter with RVTP, except that the first sequence needs rather long time (about 13s), others can all be compelled in about 4s, so the purpose of detecting in real time is achieved.

Tab.1 Sequence's Hurst parameter and its computational time under normal traffic

Sequences	Hurst parameter	Computational time(s)
a	0.812	13
b	0.792	4.0
c	0.786	3.8
d	0.808	3.9

2. Testing the sequences containing DDoS attack traffic with different intensity in real time

To get a clear understanding of how the DDoS attack traffic influences the Hurst parameter, we test the sequences containing DDoS attack traffic with different intensity. When analyzing the data of this experiment, in order to get precise experimental values, every renewed data's number h is 5% of the total sequence's number. In Fig.3, we can see the DDoS traffic's proportions in every sequence: A is 0% (A is a normal network traffic sequence); B is 5%; C is 10%; $\cdots$; T is 95%; U is 100%. Through increasing DDoS attack traffic's proportions gradually, we can see directly the changes of the Hurst parameter in Fig.3.

Beginning from the curve of Sequence A, along with the gradual increase of DDoS attack traffics, the Hurst parameter is becoming bigger and bigger. What can also be observed is that, the 5% and 10% DDoS attack traffics in the beginning of the process influence the Hurst parameter greatly. Along with the increasing of DDoS attack traffic, its influence on the Hurst parameter decreases gradually. As showed in Tab.2, when the proportion of DDoS traffic increases from 10% to 95%, the Hurst parameter only varies from 0.975 to 0.998, which indicates that the increment of the Hurst parameter is very slow. When the overwhelming majority of the network traffic is DDoS attack traffic, the Hurst parameter decreases along with the further increment of DDoS attack traffic.

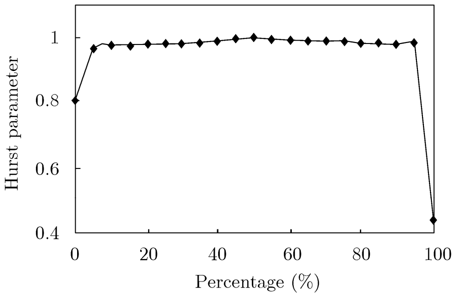

Fig.3 The change rule of the Hurst parameter of DDoS attack traffics with different intensity

Tab.2 The Hurst parameters of sequences (Seqs.) containing DDoS attack traffics with different intensity

Seqs.	A	B	C	D	E	F	G
H	0.812	0.965	0.976	0.977	0.978	0.981	0.983
Seqs.	H	I	J	K	L	M	N
H	0.985	0.989	0.994	0.998	0.995	0.991	0.988
Seqs.	O	P	Q	R	S	T	U
H	0.987	0.986	0.985	0.981	0.979	0.975	0.443

V. Real-time Detection of DDoS Attack Based on Fuzzy Logic (FRVTP)

Traditional DDoS judgment mainly depends on the known Hurst parameter and experience; it lacks self-adaptability and has much subjectivity. DDoS attack is a process that changes dynamically and frequently. We introduce fuzzy logic judgment for real-time detection of DDoS attack; it improves the self-adaptability and the agility of DDoS judgment.

1. Fuzzy logic judgment idea

Fuzzy logic judgment disposes information based on fuzzy or non-fuzzy reasoning rules[13]. It makes self-adaptive judgment in light of mature experience. The general fuzzy judgment process is shown in Fig.4. It consists of three parts: (1) Fuzzy quantitative disposal, it makes the real input parameter as a fuzzy set. (2) Fuzzy judgment rules, *i.e.* knowledge base. It includes the definition of fuzzy set and fuzzy operators. (3) Fuzzy decision, *i.e.* reasoning mechanism. It carries out all the output calculation.

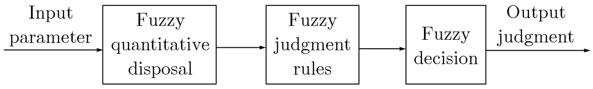

Fig.4 The general fuzzy judgment process

2. The implement mechanism of intelligent fuzzy logic introduction

As is shown in Fig.5, the structure of fuzzy judgment is two-dimensional input and one-dimensional output[14]. The two inputs are the formal Hurst parameter and its changing speed. They reflect not only the influence of the Hurst parameter on DDoS attack degree, but also the influence of the Hurst parameter's real-time changing tendency on that. Consequently, the self-adaptability and accuracy in real time have been greatly improved.

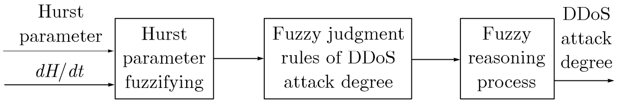
Hurst parameter
dH/dt
Hurst parameter fuzzifying
Fuzzy judgment rules of DDoS attack degree
Fuzzy reasoning process
DDoS attack degree

Fig.5 DDoS judgment process based on fuzzy reasoning

DDoS attack process is a complex and sensitive system. When selecting fuzzy domain and fuzzy subset, this process adds the number of elements in order to distribute them in reason, thereby cover the whole domain rationally. Define the changing scope of the Hurst parameter as the domain of fuzzy set: H={$-5,-4,-3,-2,-1,0,1,2,3,4,5$}; define the changing scope of HC (Hurst parameter's changing speed) as the domain of fuzzy set: HC={$-5,-4,-3,-2,-1,0,1,2,3,4,5$}, the fuzzy subsets are $\overline{H}$={NB,NM,NS,ZO,PS,PM,PB}, $\widetilde{\text{HC}}$={NB,NM,NS,ZO,PS,PM,PB}, respectively, where "NB" stands for the negative big, "NM" the negative middle, "NS" the negative small, "ZO" the zero, "PS" the positive small, "PM" the positive middle, "PB" the positive big. Define the fundamental universe of H as [H_s, H_e], and the fundamental universe of HC as [HC$_s$, HC$_e$], we get the fuzzy judgment rules of the DDoS attack degree. Fuzzy judgment result is variable L. Fuzzy set of $\widetilde{L}$ is shown as {NA，LA，MA，SA，CA}, where "NA", "LA", "MA", "SA", and "CA" represent no DDoS attack, light DDoS attack, medium DDoS attack, severe DDoS attack and entire DDoS attack, respectively. The variable's membership degree function of each fuzzy language is normal school $\mu(x)=e^{-((x-a)/b)^2}$. We can get the membership degreed assignments of every fuzzy subset as well as the fuzzy judgment model of every parameter. We make experiments on DDoS attack time after time, taking notes of the relationship between the Hurst parameter, changing tendency of the Hurst parameter and DDoS intensity. Consequently, we achieve the fuzzy judgment rules, the fuzzy rule of dynamic self-adaptability can be expressed as follows:

$$\left.\begin{aligned} L &= -<\beta H+(1-\beta)\mathrm{HC}> \\ \beta &= \frac{1}{4}(\beta_s-\beta_0)|H|+\beta_0 \end{aligned}\right\} \tag{8}$$

where $0\le\beta\le\beta_s\le 1,\ \beta\in[\beta_0,\beta_s]$. The fuzzy rule can adjust the control weight automatically on line according to fuzzy variable *H*. This method accords well with the thinking mechanism of decision process and is easy to be implemented. As being different from the traditional method depending on experience, it can adjust the fuzzy rule in terms of the real dynamic network environment.

The data in the experiment is as follows: The actual data scope of the Hurst parameter [H_s,H_e]= [0.4,1], the scope of the Hurst parameter's changing speed [HC$_s$, HC$_e$]=[–0.6/s,0.6/s], the membership degree function $\mu(x)=e^{-((x-a)/b)^2}$, where $a=0$, $b=1$, $\beta_s=0.8$, $\beta_0=0.35$.

3. Fuzzy self-adaptive judgment of DDoS attack based on the Hurst parameter

An intelligent DDoS judgment system is devised with fuzzy logic according to the former experiment data. In light of the basic theory and method of fuzzy mathematics, judgment rules' conditions and operation of DDoS attack degree are shown in fuzzy set. Deposit these fuzzy judgment rules and relative information as knowledge into repository. According to the actual dynamic attack process, the network element equipments implement the self-adaptive judgment of DDoS attack in real time intelligently by using fuzzy reasoning. Tab.3 shows the input parameters used in fuzzy judgment. The first one is the Hurst parameter of sampling sequence in normal traffic (sampling time is 10s). Then, we get the Hurst parameter of new sequence after the former sampling sequence of DDoS attack traffic updating respectively 10%, 20%, $\cdots$, 90%, 100% and the Hurst parameter occurred entirely after adding 10% of the normal flow data in DDoS attack.

Select the input parameter H and dH/dt in Tab.3, and put them into the two-dimensional input and one-dimensional output fuzzy judger shown in Fig.5, we get the fuzzy judgment rules of the DDoS attack degree. Fuzzy judgment result is the variable L. Fuzzy set of $\tilde{L}$ is shown as {normal, light, medium, severe, entire}, where "normal", "light", "medium", "severe", "entire" represent no DDoS attack, light DDoS attack, medium DDoS attack, severe DDoS attack and entire DDoS attack, respectively. Tab.4 shows the results.

Tab.3 Input parameter in fuzzy judgment of DDoS attack traffic

t(s)	H	dH/dt
10	0.812	0
11	0.976	0.164
12	0.978	0.002
13	0.983	0.005
14	0.989	0.006
15	0.998	0.009
16	0.991	–0.007
17	0.987	–0.004
18	0.985	–0.002
19	0.979	–0.003
20	0.443	–0.536
21	0.975	0.532

Tab.4 Fuzzy judgment result of DDoS attack degree

$\frac{dH}{dt}$	H											
	0.443	0.812	0.975	0.976	0.978	0.979	0.983	0.985	0.987	0.989	0.991	0.998
–0.536	—	entire	entire	entire	entire	entire	entire	entire	entire	entire	entire	entire
–0.007	entire	normal	severe	severe	severe	severe	severe	severe	severe	severe	severe	medium
–0.004	entire	normal	severe	severe	severe	severe	severe	severe	severe	severe	severe	medium
–0.003	entire	normal	severe	severe	severe	severe	severe	severe	severe	severe	medium	medium
–0.002	entire	normal	severe	severe	severe	severe	severe	severe	severe	severe	medium	medium
0.002	entire	normal	light	light	light	light	medium	medium	medium	medium	medium	—
0.005	entire	normal	light	light	light	medium	medium	medium	medium	medium	medium	—
0.006	entire	normal	light	light	medium	medium	medium	medium	medium	medium	medium	—
0.009	entire	normal	medium	medium	medium	medium	medium	medium	medium	medium	—	—
0.164	—	light	—	—	—	—	—	—	—	—	—	—
0.532	severe	—	—	—	—	—	—	—	—	—	—	—

Note: "—" means that this instance will not happen.

4. Comparison with the traditional detection method

In order to compare FRVTP method with the traditional methods which based on statistics and feature-packets, respectively, we carried out the following experiments. Firstly, we focus on the attack launched by the 1999 DARPA Offline Intrusion Detection Evaluation. Secondly, we adopt the DDoS attack tool made by ourselves. Thirdly, we test the accuracy of high pressure detection of Web server which is launched by Web detection tool. Tab.5 shows the experimental contrast between the detection method based on FRVTP and the traditional methods.

When carrying out the first contrast experiment, we attack the Web server using 1999 DARPA Offline Intrusion Detection Evaluation. All of the three detection methods all detect the attack successfully. The detection method based on statistics adopts the worldwide SkyNet Firewall[15]. The SkyNet Firewall reports the attack in about 3s after DDoS attack happens while FRVTP detection method reports in 10s. The longer time involve is due to the definite data sampling circle. The sample will not be representative if the circle is too short. So, in the proposed method, we spend much more time and it is absolutely acceptable.

Tab.5 Experimental contrast between the detection method based on FRVTP and the traditional methods

Contents of the experiment	Detection technology of FRVTP	Detection technology based on statistics (using SkyNet Firewall)	Detection technology based on feature packets
The attack launched by 1999 DARPA Offline Intrusion Detection Evaluation	Judge correctly in 13s	Judge correctly in 3s	Detect the attack correctly
Adopt DDoS attack tool which authorized by ourselves to attack	Judge correctly in 13s	Judge correctly in 3s	Fail to detect the attack
High pressure detection (using Web detection tool)	Judge correctly in 13s	Regard high pressure detection as DDoS attack by mistake	——

Note: "——"means that this instance will not happen.

When carrying out the second contrast experiment, we launch DDoS attack toward certain web server port using the tool made by ourselves. The SkyNet Firewall and FRVTP detection both detect the attack successfully. Time involvements are 3s and 10s respectively. However, the firewall based on feature packets detection technology does not respond[16].

When carrying out the third contrast experiment, we use web detection tool to do pressure test on the web server[17]. Three clients run the detection tool at the same time, and the number of working threads of detection tool which are set at every client is 600. They are designed to simulate the network traffic of web server when the network is busy. Taking the FRVTP method, we get the normal network traffic model. However, when using the detection method based on statistics, the result is mistaken as DDoS attack. That is to say, the traditional detection method based on statistics cannot distinguish the DDoS attack and network traffic under busy operation.

From the result of the experiment, we can conclude that FRVTP method can distinguish the DDoS attack traffic and busy operation traffic better, and has great adaptability to different DDoS attacks.

VI. Conclusions

Traditional DDoS judgment mainly depends on the known Hurst parameter and experience; it lacks self-adaptability and has great subjectivity. Aiming at overcoming this shortcoming, we put forward the VTP method to calculate the Hurst parameter using the slide-window mechanism, to detect DDoS attack in real time. This method can resolve several key problems such as parameter selection and the Hurst parameter evaluation in the process. Adopting FRVTP method, we analyze the 1999 DARPA Offline Intrusion Detection Evaluation, and then we conclude the Hurst parameter changing law of network traffic self-similarity model in the process of DDoS attack. After that, using fuzzy logic technology, we design an intelligent DDoS judgment mechanism in the light of experimental data. Based on the basic theory and methods of fuzzy mathematics, we demonstrate the judgment rules of DDoS attack degree with fuzzy set in fuzzy judger. The experiment indicates that the new method is able to not only recognize the Hurst parameter changes caused by DDoS attack with different intensity, but also detect DDoS attack in real time. Thus, it improves the self-adaptive judgment of DDoS attack, and realizes the on-line self-adaptive judgment of DDoS attack in real time intelligently.

References

[1] W. E. Leland, M. S. Taqqu, W. Willinger, *et al.* On the self-similar nature of ethernet traffic (Extended version). *IEEE/ACM Trans. on Networking*, **2**(1994)1, 1–15.

[2] V. Paxson and S. Floyd. Wide area traffic: The failure of Poisson modeling. *IEEE/ACM Trans. on Networking*, **3**(1995)3, 226–244.

[3] T. E. Ozkurt, T. Akgul, and S. Baykut. Principal component analysis of the fractional brownian motion for 0<H<0.5. Proceedings of the International conference on Acoustics, Speech and Signal Processing

(ICASSP'2006), Toulouse, France, May 21–24, 2006, vol.3, 488–491.

[4] Y. G. Kim, A. Shiravi, and P. S. Min. Congestion prediction of self-similar network through parameter estimation. Network Operations and Management Symposium, Vancouver, Canada, April 5, 2006, 1–4.

[5] Guanghui He and J. C. Hou. An in-depth, analytical study of sampling techniques for self-similar internet traffic. The 25th International Conference on Distributed Computing Systems, Columbus, OH, June 6–10, 2005, 404–413.

[6] Y. Xiang, Y. Lin, W. L. Lei, *et al.* Detecting DDoS attack based on network self-similarity. *IEE Proceeding on Communications*, **151**(2004)3, 292–295.

[7] H. F. Zhang, Y. T. Shu, and Oliver Yang. Estimation of Hurst parameter by variance-time plots. Proceedings of the IEEE Pacific Rim Conference on Communications, Computers and Signal Processing, Victoria, BC, Canada, Aug. 20–22, 1997, vol.2, 883–886.

[8] A. Popescu. Traffic self-similarity. IEEE International Conference on Telecommunications (ICT'2001), Bucharest, Romania, June 8, 2001, 20–24.

[9] D. Guo, X. Wang, and J. Zhang. Fast real-time Hurst parameter estimation via adaptive wavelet lifting. *IEEE Trans. on Vehicular Technology*, **53**(2004)7, 1266–1273.

[10] T. Hagiwara, H. Doi, H. Tode, *et al.* High-speed calculation method of the Hurst parameter based on real traffic. Proceedings of the 25th Annual IEEE Conference on Local Computer Networks, Tampa, Florida, USA, Nov. 8–10, 2000, 662–669.

[11] Information Systems Technology Group of MIT Lincoln Laboratory. The 1999 DARPA intrusion detection evaluation data set. http://www.ll.mit.edu/IST/ideval, June 18, 2006.

[12] Qin Yu, Yuming Mao, Taijun Wang, *et al.* Hurst parameter estimation and characteristics analysis of aggregate wireless LAN traffic. Proceedings of the International Conference on Communications, Circuits and Systems, Hong Kong, China, May 27–30, 2005, vol.1, 339–345.

[13] Lixin Wang and Yingjun Wang. A Course in Fuzzy Systems & Control. 1st ed. Beijing, China, Tsinghua University Press, 2003, 55–66 (in Chinese).
王立新, 王迎军. 模糊系统与模糊控制教程. 第一版. 北京, 清华大学出版社, 2003, 55–66.

[14] M. Sato and Y. Sato. Fuzzy clustering model for asymmetry and self-similarity. Proceedings of the Sixth IEEE International Conference on Fuzzy Systems, Barcelona, Spain, July 1–5, 1997, vol.2, 963–968.

[15] H. Debar and A. Wespi. Aggregation and correlation of intrusion-detection alerts. Proceedings of the 4th International Symposium on Recent Advances in Intrusion Detection, Davis, CA, October 10–12, 2001, 85–103.

[16] Y. Soejima, E. Y. Chen, and H. Fuji. Detecting DDoS attacks by analyzing client response patterns. Proceedings of the 2005 Symposium on Applications and the Internet Workshops, Saint Workshops, Italy, Jan. 31–Feb. 4, 2005, 98–101.

[17] Qiang Yang and Ke Wang. Web-log cleaning for constructing sequential classifiers. *Applied Artificial Intelligence*, **17**(2003)5, 431–441.