Early DoS Attack Detection using Smoothened Time-Series and Wavelet Analysis

Pravin Shinde, Srinivas Guntupalli
CDAC, Mumbai
{pravin,srinivas}@cdacmumbai.in

Abstract

Denial of Service(DoS) attacks are ubiquitous to computer networks. Flood based attacks are a common class of DoS attacks. DoS detection mechanisms that aim at detecting floods mainly look for sudden changes in the traffic and mark them anomalous. In this paper, we propose a method that considers the traffic in a network as a time-series and smoothens it using exponential moving average and analyzes the smoothened wave using energy distribution based on wavelet analysis. The parameters we used to represent the traffic are number of bytes received per unit time and the proportion between incoming and outgoing bytes. By analyzing the energy distribution in the wavelet form of a smoothened time-series, growth in the traffic, which is the result of a DoS attack can be detected very early. As the parameters we considered represent different properties of the network, the accuracy of the detection will be very high and with less false positives.

1 Introduction

Attacks on computer networks are becoming as common as computer networks themselves. The reasons for this are manifold. Improper designing of network protocols, designing the protocols and computing systems keeping only the genuine users in mind, expecting a responsible behavior from the users, unsafe coding practices are a few to name, which has left the current computer networks vulnerable for a wide variety of attacks and exploits.

New ways of exploiting the computer networks are being invented everyday. In the same pace, detection methodology is also expanding. Attack detection methods can be largely divided into two classes. Methods that use attack signatures, also called rule-based detection methods and methods that detect anomalies. Signature based detection techniques expect a complete knowledge of the attacks in advance and look for such behavior in the system or network. However this kind of methods can't detect attacks which are not seen earlier. Anomaly detection methods detect novel attacks as well, as long as the attack creates a significant deviation from the normal, in the data analyzed by the detection system.

The attacks on computers and computer networks can be broadly classified into two categories. First, attacks that use knowledge of the victim or victim's environment and exploit the vulnerabilities that exist in the victim. These kind of attacks need sophisticated tools and mechanisms and extensive knowledge of victim. Second, attacks that typically use brute force. By applying brute force these attacks render the resources useless for legitimate users. Some of the famous attacks come under this category. In February 2000, Yahoo became the first website hit by a series of high-profile attacks in a three-day period [1]. During the next few hours, Amazon.com, eBuy, CNN.com, Buy.com, ZDNet, E*Trade, Excite.com were all subjected to total or regional outages by DDoS attacks.

DoS attacks are mainly caused by either exploiting a vulnerability in an application or flooding a network [6] or a system with too many requests than it can handle, so that it won't be able to service further requests. In both the cases the aim of the attacker would be to make a resource unusable for the intended users of the resource. The flooded network or network service will be busy handling the requests with malicious intent and will run out of resources to serve genuine requests causing a DoS for genuine users.

In this paper we aim at detecting flood based DoS attacks by monitoring the changes in the network traffic in the form of a time series.

Rest of the paper is organized as follows. In section 2, related work has been presented. In Section 3, we talk about the approach we followed and the improvements we suggested over the method used in [11]. Section 4 explains the simulation environment we used for creation and detection of DoS attacks, followed by Results and Discussion in Section 5 and Future Work and Conclusion in Section 6.

2 Related Work

When a network is under flood based DoS attack, sudden changes in the traffic are manifested. DoS attack detection

0-7695-2876-7/07 $25.00 © 2007 IEEE
DOI 10.1109/IAS.2007.16

methods look for this kind of changes in the network and term this as anomalous.

Attack detection methods can be largely classified into methods,

- that build a model by learning the normal behavior of the network and validate the regular traffic against this model
- that find anomalies by monitoring continuous changes in the network without any assumption of normalcy. Here, sudden and significant change in the network is considered as an attack

Some of the approaches in literature consider static information like the set of IP addresses that are normally seen in a particular network as normal. And when there is heavy or suspicious traffic only those are allowed in and others are dropped[12]. These set of IP addresses are continuously updated. While building this set, IP addresses that complete three-way handshake are considered genuine as spoofing is ruled out in such cases. If an IP present in this set is not active for a certain time period, it will be moved out of the set. This way, though dynamic changes in the network usage are not allowed, attacks that use random IP addresses by spoofing can be curtailed. [9] uses hop count in the received packet to infer the genuineness of the IP address. Though attacker can spoof an IP address, hop count determined from the time-to-live gives an idea of whether it is genuine or not. Once a map of IP address ranges and expected hop count to reach a network is created, that can be used for look up and finding the potential IP address spoofs will be easy.

Methods that build a model of normal traffic often use machine learning algorithms. Clustering has been used in [4] and [3]. [4] uses BIRCH clustering algorithm. A vector with 'n' parameters is extracted from the normal traffic data and is used to create CFTrees . If the distance of the test data is more than a certain threshold, test data would be considered anomalous. Similarly, in [3] labelled data (that include both normal and anomalous data) with 'n' parameters is arranged in the form of a grid with every parameter as a dimension and clusters of normal and anomalous data are created. Test data is also distributed in the grid and is classified depending on the type of cluster it falls in. Support Vector Machines and Radial Basis Function Neural Networks are used in [14] to create binary classifies that classify data into normal and anomalous. [15] uses a multi-layer perceptron as classifier to classify a network as source or victim of a DDoS attack or normal. Input fed to the perceptron is passive information collected from traffic analysis like the ratio between the incoming and outgoing packets.

Some methods detect the sudden changes in traffic by converting the data into a time series and analyzing the time series. They analyze the time series in two different ways:

- Finding the distribution of data in a sampling period and if it is above certain threshold terming this as anomalous
- Finding the monotonous change in some parameter with time and if the change is substantial terming the data as anomalous

The rationale behind detection methods that look for sudden changes is an assumption that the proportion between certain parameters remains roughly uniform as long as traffic is normal. In [16] the parameters considered are the number of requests made and the number of responses received. During a DoS attack, the usual proportion between these parameters breaks and this is detected using Change Point detection method which detects whether the given time series is statistically homogeneous or not. Distribution of IP addresses in the network traffic has been considered as an important parameter that gets effected by DoS attacks in [17]. Chi-Square statistic [18] and covariance [10] are also used to detect statistical heterogeneity in the time series.

Wavelet analysis is able to capture complex temporal correlation across multiple time scales. [11] used energy distribution based on wavelet analysis to detect DoS attacks. When traffic behaviors are affected by DoS attack, energy distribution variance changes markedly. This change in distribution is used to detect DoS attack. We follow this approach and attempt to improve the performance.

3 Our Approach

Wavelet analysis can be used to detect the change in network traffic patterns. When a DoS attack takes place, the traffic pattern changes considerably, which can be easily detected using energy distribution analysis of wavelet that represents the traffic in the form of a time series. Energy distribution analysis of wavelets to detect DoS attacks has been used by [11].

The above approach may lead to false positives as small fluctuations in the traffic may also be termed as anomalous. The method we propose smoothens the time series before applying the energy distribution function on the time series. Small changes which are not monotonous will be smoothened and energy distribution remains unaffected.

The method we propose monitors a link or a node in the network and generates the time series of number of packets, number of bytes and the ratio between the number of incoming and the number of outgoing packets. Time is divided into discrete time windows and the three parameters are computed for each time window. Due to genuine transient changes in the traffic there might be small and significant fluctuations that might lead to a lot of variation in

the values that represent the above parameters. If we analyze this data without normalizing, that might result in lot of false positives. To handle this, we propose normalization using exponential smoothing average for each parameter. The time series of all the parameters is smoothened giving different weights to history and to the recent past. There is a possibility of reducing the impact of an attack due to this smoothing. But, as we look for the monotonous increase in values, even though small, an attack will be captured early enough.

Once we produce a time series, it is analyzed using energy distribution analysis of wavelets. The energy distribution function captures the gradual growth in the time series into a spike very early. There will be significant difference in energy distribution between consecutive time windows during the early stages of attack. This can be flagged as anomalous. The following sections explain the method we followed in detail.

3.1 Generation of a Time Series

Network traffic can be represented in the form of a time series considering some of the traffic parameters such as number of packets or number of bytes passing through a line per unit time. Apart from this, parameters like number of connection requests (in case of attacks based on TCP), proportion between incoming and outgoing packets, other protocol dependent parameters are also considered. We computed the number of bytes passing through a line closest to victim and converted the traffic into a time series considering 100 milli-seconds as unit time.

3.2 Exponential Smoothing

When the wavelet analysis is done on traffic that is represented in the form of a time series, small and sudden genuine changes in the traffic also effect the result drastically. Instead of the actual values, if moving average is considered to smoothen the changes, effect of the small aberrations can be contained. In exponential smoothing the most recent observation gets a little more weight than 2nd most recent, and the 2nd most recent gets a little more weight than the 3rd most recent, and so on. The simple exponential smoothing (x) model accomplishes this. Let α denote a "smoothing constant" (a number between 0 and 1) and let S(t) denote the value of the smoothened series at period t. The following formula is used recursively to update the smoothened series as new observations are recorded where Y(t) is the last smoothened value and S(t-1) is the current observation.

$$S(t) = \alpha Y(t) + (1 - \alpha)S(t - 1) \quad (1)$$

The value of α decides how much weight-age should be given to the history and how fast the average should be moved according to the latest changes. We used 0.3 for the α value. Another advantage of exponential smoothening is that it can be computed in real time with very less computation, and it does not depend on data of future time periods and doesn't handle too much data.

3.3 Energy Distribution

Once the smoothened time series data is collected, this data can be represented in the form of a wavelet. Wavelet uncouples the scaled traffic into several components. Energy distribution among these components brings out the difference between neighboring components and correlation among them.Energy distribution function decomposes the original wave form into approximation and detail. Approximation can be decomposed further recursively into second level of approximation and detail. This process can be done multiple times to generate wavelet decomposition tree.

Energy distribution function takes the decomposition tree and computes the distribution of energy among different components. When the variation in the time series is less, the distribution of energy in a time window will not change significantly with time. When the traffic is anomalous, time series will capture the anomaly and the same will appear as variation in the energy distribution. If we observe the difference in distribution of energy between two consecutive time windows, it will be high when the traffic is anomalous.

The energy distribution is applied on the smoothened data, by creating time windows of fixed size. Each time window contains a fixed number of time slots. Window size is one of the critical parameters in the detection of the attack. If it is small then it will raise false alarms on small fluctuations. If it is very large then it may miss the fluctuations due to malicious data. The windows are created with a time overlap between two consecutive windows, so that the growth of traffic is captured right at the beginning of the attack. The amount of overlap is also an important parameter, as too much overlap will absorb the effect of variation and too less overlap may not show the variation at all.

The energy distribution for each window is calculated and the difference between the energy distribution of two consecutive windows is considered for the detection of the attack. If the variation in the energy distribution is constantly increasing for the last few windows then it can be considered as beginning of an attack. [11] uses a threshold for the difference of energy distribution in two consecutive time windows to detect an attack. There is a possibility of false positives with this approach as even when there is a variation in energy distribution between only two windows in the entire time series, it will be flagged as an attack. This can be improved by observing the difference in energy dis-

tribution for few time windows in a series and if the distribution changes and remains constant for a few time windows, it indicates the gradual growth which is possibly the result of an attack. If the distribution changes and again comes back to normal, that might be due to a flash event.

4 Simulation Environment

We have created networks of 100 and 200 nodes using NS2 (Network Simulator) [2]. The simulation includes three level hierarchical network which was created using GT-ITM Topology Generator [5]. All nodes in the network will be participating in the generation of traffic. The traffic generated is CBR application based. The attack simulated is an UDP flood attack, which is common when attacking DNS servers.

The simulation was repeated with various configuration for network size, simulation time, number of attacks, number of attacking machines. Two hierarchical networks of size 100 and 200 were created for purpose of simulations. The attack was simulated by randomly selecting a few nodes (10 to 15) as attacker nodes and attacking one machine by creating many flows and sending many packets to victim machine. The logs on the bottleneck link were used for detection of attack. Simulations with one and two attacks in given time interval were conducted. Simulation time was varied from 500 seconds to 6000 seconds to simulate attacks with various intensities.

5 Results and Discussion

Fig.1 shows the attack scenario. Duration of simulation is 6000 seconds. Attack starts at 3200 seconds and continues up-to 5000 seconds. From the figure it is evident that there is a peak at 4500. Attack should be detected much before that peak to take preventive actions against the attack.

5.1 Early Detection

Though the attack started after 3200 seconds, traffic peak is observed only at 4500 seconds. If the data is analyzed looking for peaks in the traffic, detection would have been possible only after 4000 seconds. But, analysis of time series using the energy distribution variation in the consecutive time windows detects the attack around 3700 seconds, much before the peak of attack, which is evident in Fig.1. Third part of Fig.1 shows the graph of energy distribution variation with SES, where the attack can be detected earlier compared to the second part of Fig.1.

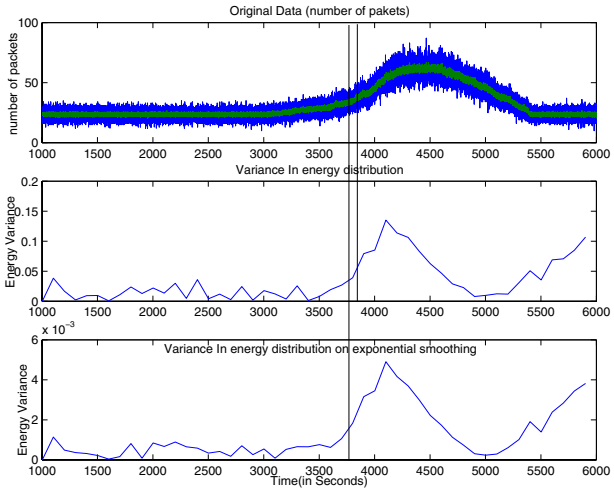

Figure 1. Early Detection

5.2 Effect of smoothing for different Window Sizes

Window size considered for finding the energy distribution is very critical. In Fig.2 and Fig.3 the window sizes used are 1000 and 600 seconds respectively. These graphs show the variance in energy distribution of consecutive windows. The difference in energy distribution in consecutive windows is plotted against the time. The window size for these graphs is 1000 seconds with overlapping of 1/10th of window size. When the window size is large , analysis of data with moving average and without moving average gave same results which can be observed in Fig.2. But detection of the attack can happen only at around 4000 seconds. To make a little earlier detection possible we should go for a smaller time window. In Fig.3, a smaller time window is considered. The window size for these graphs is 600 seconds with overlapping of 1/10th of window size. The result of the analysis here shows some false positives in case of data without moving average, as data that is not smoothened shows large variation even for a small fluctuation. But, smoothened data absorbs these small fluctuations and shows monotonous growth only when there was an actual attack.

5.3 Ratio between in-coming and out-going packets

There might be some false positives when the above method alone is used for analysis. For example during some flash events, the traffic suddenly goes up and the destination nodes will be able to respond to the sudden growth in requests. But there will be significant change in the energy distribution when the time-series corresponding to the traf-

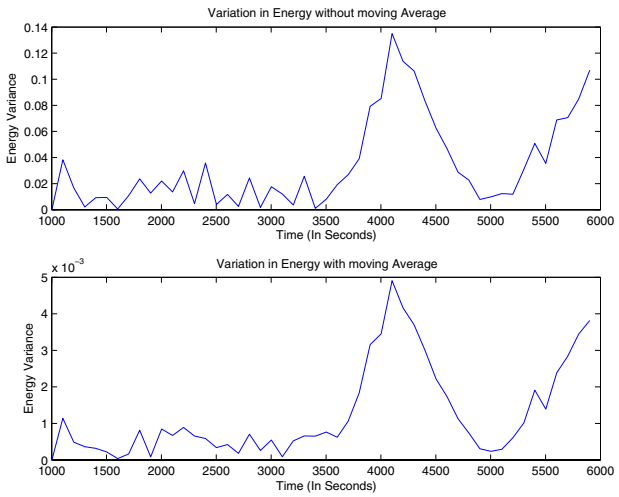

Figure 2. Analysis with a larger time window (1000 seconds)

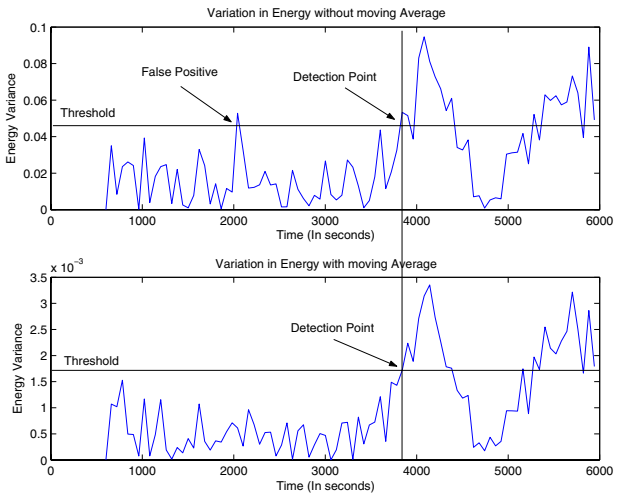

Figure 3. Analysis with a smaller time window (600 seconds)

fic is analyzed which will be flagged as anomalous. To reduce these false positives, [16] proposed usage of proportion of in-coming and out-going packets as a parameter that reflects DoS attack. We improve this by analyzing the same parameter using energy distribution variation of the proportion between in-coming and out-going traffic. If the volume of traffic increases but server is able to reply to all the requests, then the ratio of in-coming and out-going packets will not change much. Only if server is not able to respond to many requests, then this ratio will change as incoming traffic will increase considerably over outgoing traffic. So this parameter can reduce false positives when volume of traffic is increased, but server is able to reply to all the requests. In Fig.4, the volume of traffic is increased, but server is responding to all requests. So though variation in energy distribution of number of packets is showing a spike, but variation in energy distribution of ratio of in-coming to out-going packets is not showing the similar spike, stating that it is just increase in volume of traffic, but not an attack, thus reducing false positives.

The method we used can analyze the data as it arrives, so it can work in near real time. The response time is decided by the parameters like window size and the extent of overlap of windows. This method used more than one parameter and that increases the confidence of the decision. If we use just one parameter, the case with many of the existing solutions, that parameter might reflect a specific situation very well, but when it comes to handling real life data, it might lead to many false positives. If the attacker is aware of the parameter that is being used, there might be some ways to circumvent and realize the attack. But when many parameters capture the behavior, it would be difficult to circumvent all of them.

In our simulation we captured the behavior of the network using three parameters, number of packets, number of bytes and proportion between number of in-coming and out-going packets.

5.4 Performance

Table.1 and Table.2 show the behavior of the proposed method for various lengths of time windows and different overlappings between consecutive time windows. Time windows considered are 100, 150 and 200 seconds with overlapping of 50%, 80% and 90%. Table.1 shows the results without SES and Table.2 with SES. 10 attacks had been done during the simulation. Number of true and false positives are represented as N/M where N is the number of true positives and M is the number of false positives. It is evident from the tables that analysis after SES gives better results, both in terms of reducing false positives and increasing true positives.

Overlap	50%	80%	90%
100	8/1	9/1	9/1
150	8/1	8/2	8/1
200	6/1	8/0	10/1

Table.1 Results without performing SES

Overlap	50%	80%	90%
100	9/0	10/0	9/1
150	9/1	10/1	9/1
200	8/1	9/1	9/1

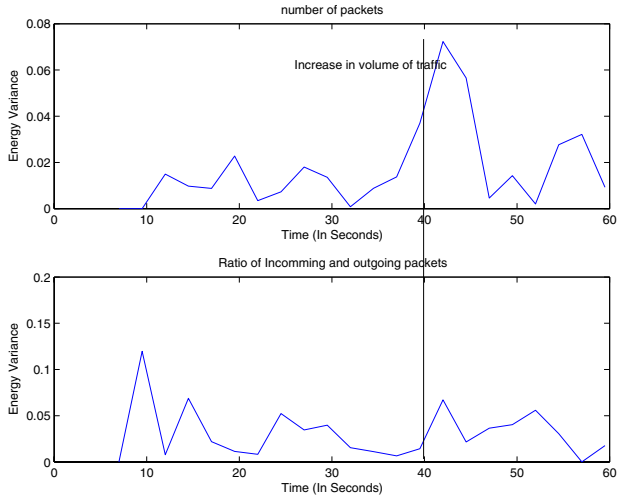

Figure 4. Analysis using proportion between in-coming and out-going packets

Table.2 Results after performing SES

6 Conclusion and Future Work

Many techniques that are prevalent for detection of DoS attack look for a sudden spike in network traffic or traffic destined to a particular host. In this paper we proposed a method that uses multiple parameters viz., number of packets, number of bytes and proportion between number of incoming and out-going packets and analyzed the corresponding time series using energy distribution in wavelets. We could reduce false positives by smoothing the time series before analysis, as it absorbs flash events. And using multiple parameters together gives more confidence to the result.

Clubbing the result of this method with the information obtained from network management and monitoring systems can improve the confidence and paves way for taking preventive actions against the attack.

7 Acknowledgements

We thank Dr.Sasi Kumar, C-DAC, Mumbai, Mr.Abhishek Seth, IIT Bombay, Mumbai and our colleagues for their constant guidance, support and encouragement and CDAC for providing the infrastructure to carry out this work.

References

[1] The denial-of-service aftermath. In *http://archives.cnn.com/2000/TECH/computing/02/14/dos.aftermath.idg/index.html*.

[2] The network simulator - ns-2. In *www.isi.edu/nsnam/ns/*.

[3] S. K. Bethi, V. V. Phoha, , and Y. B. Reddy. Clique clustering approach to detect denial-of-service attacks. In *5th Annual IEEE Information Assurance Workshop*, June 2004.

[4] K. Burbeck and S. Nadjm-Tehrani. Adwice - anomaly detection with real-time incremental clustering. In *7th International Conference in Information Security and Cryptology, ICISC*, pages 407–424, 2004.

[5] K. Calvert and E. W. Zegura. Gt-itm: Georgia tech internetwork topology models. 1997.

[6] G. Carl and G. Kesidis. Denial-of-service attack detection techniques. In *IEEE Internet Computing*, pages 82–89, Feb. 2006.

[7] T. M. Gil and M. Poletto. Multops: a data-structure for bandwidth attack detection. In *Proceedings of 10th Usenix Security Symposium*, Aug. 2001.

[8] J. Haggerty, T. Berry, Q. Shi, and M. Merabti. Diddem: A system for early detection of tcp syn flood attacks. In *IEEE Communications Society, Globecom 2004*, 2004.

[9] C. Jin, H. Wang, and K. G. Shin. Hop-count filtering: An effective defense against spoofed ddos traffic. In *Proceedings of the 10th ACM conference on Computer and communications security*, pages 30–41, 2003.

[10] S. Jin and D. S. Yeung. A covariance analysis model for ddos attack detection. In *IEEE Communications Society*, 2004.

[11] L. Li and G. Lee. Ddos attack detection and wavelets. In *ICCCN 2003. Proceedings of The 12th International Conference on Computer Communications and Networks*, pages 421–427, Oct. 2003.

[12] T. Peng, C. Leckie, and R. Kotagiri. Protection from distributed denial of service attacks using history-based ip filtering. In *ICC '03. IEEE International Conference on Communications*, pages 482–486, May 2003.

[13] T. Peng, C. Leckie, and K. Ramamohanarao. Detecting distributed denial of service attacks using source ip address monitoring. In *IEEE Infocom, Hongkong*, 2004.

[14] G. C. Sang, P. P. Chan, D. S. Yeung, and E. C. Sang. Denial of service detection by support vector machines and radial-basis function neural nework. In *Proceedings of 2004 International Conference on Machine Learning and Cybernetics*, pages 4263–4268, Aug. 2004.

[15] C. Siaterlis and V. Maglaris. Detecting incoming and outgoing ddos attacks at the edge using a single set of network characteristics. In *ISCC*, pages 469–475, 2005.

[16] H. Wang, D. Zhang, and S. K.G. Change-point monitoring for the detection of dos attacks. In *IEEE Transactions on Dependable and Secure Computing*, pages 193–208, Oct. 2004.

[17] Y. Xu. Statistically countering dos. In *IEEE 2005*, 2005.

[18] N. Ye and Q. Chen. An anomaly detection technique based on a chi-square statistic for detecting intrusions into information systems. In *Quality and Reliability Engineering International*, pages 105 – 112, Mar. 2001.