International Journal of Modern Engineering Research (IJMER)
www.ijmer.com Pp-101-105 ISSN: 2249-6645

Protection against Denial of Service Attacks: Attack Detection

[1]P.Babu Prakash Kumar, [2]Ashish Umesh Shah

[1] Faculty, Department of Computer Science and Engineering, AVIT
[2] Students, CSE, AVIT

ABSTRACT: *Denial of Service (DoS) is a prevalent threat in today's networks because DoS attacks are easy to launch, while defending a network resource against them is disproportionately difficult. Despite the extensive research in recent years, DoS attacks continue to harm, as the attackers adapt to the newer protection mechanisms.*

There is an emerging need for the traffic processing capability of network security mechanisms, such as intrusion detection systems (IDS), to match the high throughput of today's high-bandwidth networks. Recent research has shown that the vast majority of security solutions deployed today are inadequate for processing traffic at a sufficiently high rate to keep pace with the network's bandwidth. To alleviate this problem, packet sampling schemes at the front end of network monitoring systems (such as an IDS) have been proposed. However, existing sampling algorithms are poorly suited for this task especially because they are unable to adapt to the trends in network traffic. Satisfying such a criterion requires a sampling algorithm to be capable of controlling its sampling rate to provide sufficient accuracy at minimal overhead. To meet this utopian goal, adaptive sampling algorithms have been proposed. In this paper, we put forth an adaptive sampling algorithm based on weighted least squares prediction. The proposed sampling algorithm is tailored to enhance the capability of network based IDS at detecting denial-of- service (DoS) attacks. Not only does the algorithm adaptively reduce the volume of data that would be analyzed by an IDS, but it also maintains the intrinsic self-similar characteristic of network traffic. The latter characteristic of the algorithm can be used by an IDS to detect DoS attacks by using the fact that a change in the self-similarity of network traffic is a known indicator of a DoS attack.

I. INTRODUCTION

A Denial of Service attack (DoS) is any intended attempt to prevent legitimate users from reaching a specific network resource. Such attacks have been known to the network research community since the early 1980s..

Denial of Service (DoS) attacks are undoubtedly a very serious problem in the Internet, whose impact has been well demonstrated in the computer network literature. The main aim of a DoS is the disruption of services by attempting to limit access to a machine or service instead of subverting the service itself. This kind of attack aims at rendering a network incapable of providing normal service by targeting either the network_s bandwidth or its connectivity. These attacks achieve their goal by sending at a victim a stream of packets that swamps his network or processing capacity denying access to his regular clients. In the not so distant past, there have been some large-scale attacks targeting high profile Internet sites.

DoS/DDoS attacks are a strong, comparatively new type of Internet attacks, they have basis some Biggest web sites on the world -- owned by the mainly famous E-Commerce companies such as Yahoo, eBay, Amazon -- became unreachable to customers, partners, and users; the financial losses are very huge. While former security threats could be faced by a tight security policy and active measures like using recalls, vendor patches etc. these DDoS are novel in such way that there is no totally pleasing protection yet.

II. DOS ATTACKS

2.1. Defining DoS attacks

According to the WWW Security FAQ [4] a DoS attack can be described as an attack designed to render a computer or network incapable of providing normal services. A DoS attack is considered to take place only when access to a computer or network resource is intentionally blocked or degraded as a result of malicious action taken by another user. These attacks don_t necessarily damage data directly or permanently, but they intentionally compromise the availability of the resources.

The most common DoS attacks target the computer network_s bandwidth or connectivity. Bandwidth attacks flood the network with such a high volume of traffic that all available network resources are consumed and legitimate user requests cannot get through, resulting in degraded productivity. Connectivity attacks flood a computer with such a high volume of connection requests, that all available operating system resources are consumed, and the computer can no longer process legitimate user requests.

2.2. DoS attack classification

DoS attacks can be classified into five categories based on the attacked protocol level, as illustrated in Fig.

C. Douligeris, A. Mitrokotsa / Computer Networks 44 (2004) 643–666

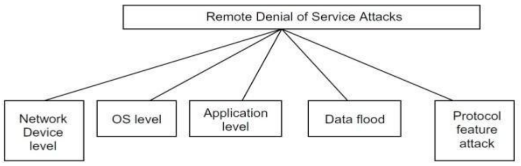

Fig. 1. Classification of Remote Denial of Service attacks.

DoS attacks in the Network Device Level include attacks that might be caused either by taking advantage of bugs or weaknesses in software, or by trying to exhaust the hardware resources of network devices. One example of a network device exploit is the one that is caused by a buffer overrun error in the password checking routine. Using these exploits certain Cisco 7xx routers [6] could be crashed by connecting to the routers via telnet and entering extremely long passwords.

In the OS level DoS attacks take advantage of the ways operating systems implement protocols. One example of this category of DoS attacks is the Ping of Death attack [7]. In this attack, ICMP echo requests having total data sizes greater than the maximum IP standard size are sent to the targeted victim. This attack often has the effect of crashing the victim_s machine. Application-based attacks try to settle a machine or a service out of order either by taking advantage of specific bugs in network applications that are running on the target host or by using such applications to drain the resources of their victim. It is also possible that the attacker may have found points of high algorithmic complexity and exploits them in order to consume all available resources on a remote host. One example of an applicationbased attack is the finger bomb [8]. A malicious user could cause the finger routine to be recursively executed on the hostname, potentially exhausting the resources of the host. In data flooding attacks, an attacker attempts to use the bandwidth available to a network, host or device at its greatest extent, by sending massive quantities of data and so causing it to process extremely large amounts of data. An attacker could attempt to use up the available bandwidth of a network by simply bombarding the targeted victim with normal, but meaningless packets with spoofed source addresses. An example is flood pinging. Simple flooding is commonly seen in the form of DDoS attacks, which will be discussed later. DoS attacks based on protocol features take advantage of certain standard protocol features.

For example several attacks exploit the fact that IP source addresses can be spoofed. Several types of DoS attacks have focused on DNS, and many of these involve attacking DNS cache on name servers. An attacker who owns a name server may coerce a victim name server into caching false records by querying the victim about the attacker_s own site.

III. DEFENCE MECHANISMS AGAINST DENIAL OF SERVICE

The extreme diversity of DoS attacks has produced similarly diverse protection proposals from the network security research community. In most cases a complete protection architecture should include the following elements: Detection of the existence of an attack.

The detection can be either anomaly-based or signature-based, or a hybrid of these two. In anomaly-based detection, the system recognises a deviation from the standard behaviour of its clients, while in signature-based it tries to identify the characteristics of known attack types. Classification of the incoming packets into valid (normal packets) and invalid (DoS packets). As in detection, one can choose between anomaly-based and signature-based classification techniques. Response. In the most general sense, the protection system either drops the attacking packets in a timely fashion or renders them harmless by redirecting them into a trap for further evaluation and analysis.

Detection and Classification usually overlap, since the method used to detect the existence of an attack often provides the necessary information to start responding towards probable normal and probable DoS traffic. Also, all three elements of protection may benefit by the use of an additional secondary element, which is the traceback of the real source of the traffic.

3.1. PROPOSED WORK

Traffic measurement and monitoring serves as the basis for a wide range of IP network operations and engineering tasks such as trouble shooting, accounting and usage profiling, routing weight configuration, load balancing, capacity planning, etc. Traditionally, traffic measurement and monitoring is done by capturing every packet traversing a router interface or a link. With today's high-speed (e.g., Gigabit or Terabit) links, such an approach is no longer feasible due to the excessive overheads it incurs on line-cards or routers. As a result, packet sampling has been suggested as a scalable alternative to address this problem.

Early packet sampling algorithms assumed that the rate of arrival of packets in a network would verage out in the long term. However, it has been shown [15] that network traffic exhibits periodic cycles or trends. The main observation of [15] and other studies have been that not only does network traffic exhibit strong trends in the audit data but these trends also tend to be long term. This section presents the proposed sampling algorithm. In Section III-A, we describe the weighted least squares predictor that is utilized for predicting the next sampling interval. This predictor has been adopted because of its capability to follow the trends in network traffic. Thereafter, in Section III-B we describe the sampling algorithm itself.

A. Weighted Least Square Predictor Let us assume that the vector Z holds the values of the N previous samples, such that ZN is the most recent sample and Z1 is the oldest sample. Having fixed a window size of N, when the next sampling occurs, the vector is right shifted such that ZN replaces ZN−1 and Z1 is discarded. The weighted prediction model therefore predicts the value of ZN given ZN−1, ...,Z1. In eneral, we can express this predicted value as a function of the N past samples i.e.,

$$\hat{Z}_N = \alpha^T \hat{Z}$$

where $\hat{z}_N$ is the new predicted value, $\hat{z}$ is the vector of past N − 1 samples, and αT is a vector of predictor coefficients distributed such that newer values have a greater impact on the predicted value $\hat{z}_N$. A second vector, t, records the time that each sample is taken and is shifted in the same manner as Z. The objective of the weighted prediction algorithm is to find an appropriate coefficient vector, αT , such that the following summation is minimized

, $S = \sum_{i=1}^{N-1} w_i \left(Z_i - \hat{Z}_i\right)^2$

where wi, Zi, and $\hat{Z}_i$ denote the weight, the actual sampled value, and the predicted value in the ith interval, respectively. The coefficient vector is given by:

$$\propto^T = (\tilde{Z}^T W \tilde{Z})^{-1} \tilde{Z}^T W$$,

where W = w^{T}w is a (N − 1) × (N − 1) diagonal weight matrix and w is a N × 1 weight vector with weight coefficient's wi that are determined according to two criteria:

1) The "freshness" of the past N − 1 samples. A more recent sample has a greater weight.
2) The similarity between the predicted value at the beginning of the time interval and the actual value. The similarity between the two values is measured by the distance between them. The smaller the Euclidean distance is, the more similar they are to each other. Based on the above two criteria, we define a weight coefficient as

$$w_i = \frac{1}{(t_N - t_i)} \left(\frac{1}{|z_i - \hat{z}_i|^2 + \eta}\right)$$, 1 ≤ i ≤ N-1,

where η is a quantity introduced to avoid division by zero.

B. Adaptive Weighted Sampling

Adaptive sampling algorithms dynamically adjust the sampling rate based on the observed sampled data. A key element in adaptive sampling is the prediction of future behavior based on the observed samples. The weighted sampling algorithm described in this section utilizes the weighted least squares predictor (see section III-A) to select the next sampling interval. Inaccurate predictions by the weighted least squares predictor indicates a change in the network traffic behavior and requires a change in the sampling rate. The proposed adaptive sampling algorithm consists of the following steps (see Fig. 1):

1) Fix the first N sampling intervals equal to τ . (In our simulations we used τ = 60 sec. and N = 10)
2) Apply the weighted least squares predictor to predict the anticipated value, $\hat{Z}_N$, of the network parameter.
3) Calculate the network parameter value at the end of the sampling time period.
4) Compare the predicted value with the actual value.
5) Adjust sampling rate according to the predefined rule set if the predicted value differs from the actual value The predicted output $\hat{\tilde{Z}}_N$which has been derived from the previous N samples, is then compared with the actual value of the sample, ZN. A set of rules is applied to adjust the current sampling interval, TCurr = tN − tN−1, to a new value, TNew, which is used to schedule the sampling query. The rules used to adjust the sampling interval compare the rate of change in the predicted sample value, $\hat{Z}$N −ZN−1, to the actual rate of change, ZN −ZN−1. The ratio, R, between the two rates is defined as:

. (5)

$$\Delta T_{New} = \begin{cases} (1+R)\,\Delta T_{Curr} \ \ if\ R > R_{MAX} \\ \beta 1 \times \Delta T_{Curr}\ \ if\ R_{MIN} < R < R_{MAX} \\ R \times \Delta T_{Curr}\ if\ R < R_{MIN}, \\ \beta 2 \times \Delta T_{Curr}\ \ if\ R \text{ is Undefined} \end{cases}$$

Based on the value of R, which ranges from RMIN to RMAX 1, we define the next sampling interval TNew as shown in Equation (6). The variables β1 and β2, in Equation 6, are tunable parameters. When determining the values for β1 and β2, one needs to consider the rate of change of the network parameter under consideration. As in [16], we used the values β1 = 2 and β2 = 2 in our simulations.

(6) The value of R is equal to 1 when the predicted behaviour is the same as the observed behavior. If the value of R is greater than RMAX, it implies that the measured value is changing more slowly than the predicted value and this means that the sampling interval needs to be increased. On the other hand, if R is less than Rmin, it implies that the measured value of the network parameter is changing faster than the predicted value. This indicates more network activity than predicted, so the sampling interval should be decreased to yield more accurate values for future predictions of the network 1 Based on the results obtained from simulations performed by us, we selected a value of RMIN = 0.82 and RMAX = 1.21. These values were selected because they provided good performance over a wide range of traffic types. parameter. The value of R may be undefined. This case arises when both the numerator and denominator of Equation (5) are zero. This condition is generally indicative of an idle network or a network in steady state. In such a scenario, the sampling interval is increased by a factor of β2(> 1).

4.2. SIMULATION RESULTS

Simulations were conducted to evaluate the performance of the proposed adaptive sampling algorithm. We evaluated the proposed sampling algorithm using data from the Widely Integrated Distributed Environment (WIDE) project [17]. The WIDE backbone network consists of links of various speeds, from 2Mbps CBR (Constant Bit Rate) ATM up to 10 Gbps Ethernet. The WIDE dataset we analyzed consisted of a 24- hour trace that was collected on September 22, 2005.

When comparing the performance of the proposed adaptive sampling algorithm with the simple random sampling

algorithm, a useful criterion to use is the mean square error (MSE) of the estimate or its square root, the root mean squared error, measured from the population that is being estimated. Formally we can define the mean square error of an estimator X of an unobservable parameter θ as
MSE (X) = E $[(X - \theta)^2]$.

The root mean square error is the square root of the mean square error and the root mean square error is minimized when θ = E (X) and the minimum value is the standard deviation of X. In Fig. 2, we compare the proposed adaptive sampling scheme with the simple random sampling algorithm using the standard deviation of packet delay as the comparison criterion. Packet delay is an important criterion for detecting DoS attacks, especially attacks that focus on degrading the quality of service in IP networks [18]. The results show that over different block sizes, the proposed adaptive scheme has a lower standard deviation when compared with the simple random sampling algorithm. Since standard deviation is directly proportional to the root mean square error criterion, this implies that the proposed algorithm predicts the packet mean delay better than the simple random sampling algorithm while reducing the volume of traffic. In the second set of experiments, we verified whether the traffic data sampled by the proposed sampling scheme has the self similar property. For this verification, we used two different parameters: the mean of the packet count and the Hurst parameter. The peak-to-mean ratio (PMR) can be used as an indicator of traffic burstiness. PMR is calculated by comparing the peak value of the measure entity with the average value from the population. However, this statistic is heavily dependent on the size of the intervals, and therefore may or may not represent the actual traffic characteristic. A more accurate indicator of traffic burstiness is given by the Hurst parameter.

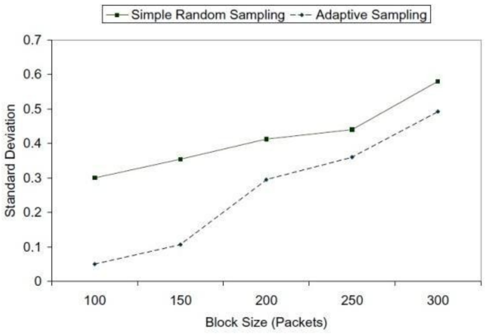

Fig. 2: Standard deviation of packet delay.

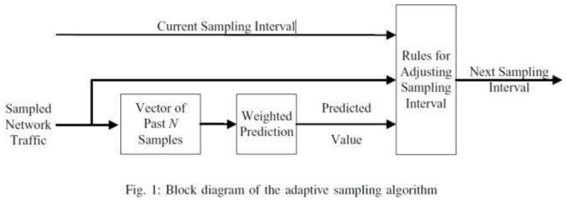
Current Sampling Interval
Rules for Adjusting Sampling Interval
Next Sampling Interval
Sampled Network Traffic
Vector of Past N Samples
Weighted Prediction
Predicted
Value
Fig. 1: Block diagram of the adaptive sampling algorithm

Fig. 3: Average percentage error for the Hurst parameter.

Fig. 3 and Fig. 4 show the average sampling error for the Hurst parameter and the sample mean, respectively. As one can see from Fig. 3, the random sampling algorithm resulted in higher average percent error for the Hurst parameter when compared to adaptive sampling. This could be the result of missing data spread out over a number of sampling intervals. In Fig. 4, the average percentage error for the mean statistic was marginally higher for our sampling algorithm when compared with the simple random sampling algorithm, albeit the difference was insignificant. One possible reason for this marginal difference is the inherent adaptive nature of our sampling algorithm—i.e., the proposed sampling algorithm is more likely to miss short bursts of high network activity in periods that typically have low network traffic. The simple random sampling scheme would be less likely to have the same problem.

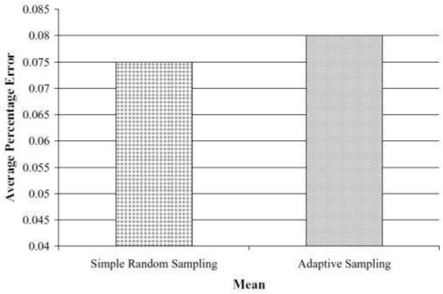

Fig. 4: Average percentage error for the mean statistic.

IV. CONCLUSION

This Paper discusses the DOS attack in and its prevention approach. In this paper, I have presented an adaptive sampling algorithm which uses weighted least squares prediction to dynamically alter the sampling rate based on the accuracy of the predictions. Our results have shown that compared to simple random sampling, the proposed adaptive sampling algorithm performs well on random, bursty data. Our simulation results show that the proposed sampling scheme is effective in reducing the volume of sampled data while retaining the intrinsic characteristics of the network traffic.

We believe that the proposed adaptive sampling scheme can be used for a variety of applications in the domain of network monitoring and network security. The variations in the self similarity and long range dependence of network traffic are known indicators of a denial-of-service attack.

Therefore, an anomaly detection scheme could successfully use the proposed sampling algorithm to sample and reduce the volume of inspected traffic while still being able to detect minor variations in the self-similarity and long range dependence of network traffic.

REFERENCES

[1] K. C. Claffy, G. C. Polyzos, and H.-W. Braun, "Application of sampling methodologies to network traffic characterization," in SIGCOMM '93: Proceedings of the Conference on Communications architectures, protocols and applications, (New York, NY, USA), pp. 194–203, ACM Press, 1993. C. NetFlow, "CISCO NetFlow." http://www.cisco.com/en/US/products/ps6601/products_ios_protocol_group_home.html.

[2] E. Millard, "Internet attacks increase in number, severity." http://www.toptechnews.com/news/Internet-Attacks-Increase-in-Severity/story.xhtml?story_id=0020007B77EI, 2005.

[3] M. Li, W. Jia, and W. Zhao., "Decision analysis of network based intrusion detection systems for denial-of-service attacks.," in Proceedings of the IEEE Conferences on Info-tech and Info-net, vol. 5, Dept. Of Computer Sci., City Univ. of Hong Kong, China, IEEE, October 2001.

[4] P. Owezarski, "On the impact of DoS attacks on internet traffic characteristics and QoS," in ICCCN '05: Proceedings of the 14th International Conference on Computer Communications and Networks, pp. 269–274, LAAS-CNRS, Toulouse, France, IEEE, October 2005.

[5] Asosheh, A., Dr. and Ramezani, N. (2008) A comprehensive taxonomy of DDoS attacks and defense mechanism applying in a smart classi¯cation. WSEAS Transactions on Computers, 7, 281{290.