Detection of Denial of Service Attacks against Domain Name System Using Machine Learning Classifiers

Samaneh Rastegari, M. Iqbal Saripan* and Mohd Fadlee A. Rasid

Abstract—Domain Name System (DNS) provides name to address mapping services for the entire chain of Internet connectivity. Hackers exploit this fact to damage different parts of the Internet. In this paper we clarify possible Denial of Service (DoS) threats against DNS. An Intrusion Detection System (IDS) is introduced is the system to detect and classify different types of DoS attacks against DNS. This system consists of a statistical preprocessor and a machine learning (ML) engine. Three different types of neural network classifiers and support vector machines are evaluated in a simulated network. The results show that a backpropagation neural network engine outperforms other types of classifiers with 99% accuracy.

***Index Terms*—Network security, Domain name system, Denial of service, Neural network, Support vector machines.**

I. Introduction

Originality DNS was designed based on an unreliable delivery protocol named User Datagram Protocol (UDP) and security of DNS was not a big issue at that point in time because the original design was sufficient to satisfy the needs of the Internet [1], [2]. Nowadays, DNS has become a vital service for the operation of the Internet and of any private network of a certain size, so this is the time to secure the DNS system from any unauthorized access.

The first objective of this paper is to evaluate different types of DoS attacks against DNS. Identifying patterns of these attacks lead us to generate the required data for different attack scenarios through simulations by varying different parameters.

Two of the most common DoS attacks occur against DNS are the type of direct DoS attacks and amplification attacks. In the first one attacker tries to overwhelm the server by sending an excess traffic from single or multiple sources. Therefore, it will cause a huge number of query packets to be received by the target name server. The name servers flooded by DoS attacks will experience packet loss and can not always respond to every DNS request. Reference [3], points that the packet size of DNS data flow is small and this similarity to anomalous packets makes the process of detection more difficult.

On the other hand, attackers establish the most sophisticated and modern type of DoS attacks known as amplification attacks to increase the effect of normal DoS attacks. The reason that this type of attack named amplification is that the attacker makes use of the fact that small queries can generate much larger UDP packets in response [4]. Nowadays, DNS protocol (RFC 2671) is used by the attackers to magnify the amplification factor. For example a 60 bytes DNS request can be answered with responses of over 4000 bytes. This yields an amplification factor of more than 60. Several researchers have studied the effects of reflected amplification attacks. Based on their analyzes, patterns of these attacks include a huge number of nonstandard packets larger than the standard DNS packet size which was 512 bytes [5].

There were several attempts to propose a solution to defend DNS against such attacks [3], [6], but according to our knowledge, there was no specific intelligent detection system for Denial of Service (DoS) threats against DNS and this is the second objective of this work.

The rest of this paper is organized as follows. Next section describes the simulation model for generating our data set. Section III introduces the proposed model for detection and classification of DoS attacks against DNS. The results are presented in section IV and, then we draw some conclusions in section V.

II. Simulation model for dataset generation

When accessing to a real environment for traffic simulation is hard, we exploit the power of network simulators. According to our knowledge, there were no available generated dataset for DoS attacks against DNS. Therefore, we used simulation for generating the required data for our experiments. We simulate our model using an OTcl program in NS-2 (version 2.28). It is used to model different DoS attacks against DNS.

The network topology of our simulation contains a single legitimate client, an attacker, and two servers. All nodes are connected to the same router. All the links are 100Mbps and 10ms except the link between target server and router that is 10Mbps and 10ms delay. We used a queue size of 100 packets, with a drop-tail queuing strategy. There are two

Manuscript received December 31, 2009.

Samaneh Rastegari is with the Department of Computer and Communication System Engineering, Universiti Putra Malaysia, 43400 Serdang, Selangor, Malaysia (e-mail: samaneh.rastegari@gmail.com).

*(Corresponding Author) M. Iqbal Saripan is with the Department of Computer and Communication System Engineering, Universiti Putra Malaysia, 43400 Serdang, Selangor, Malaysia (e-mail: iqbal@eng.upm.edu.my).

Mohd Fadlee A. Rasid is with the Department of Computer and Communication System Engineering, Universiti Putra Malaysia, 43400 Serdang, Selangor, Malaysia (e-mail: fadlee@eng.upm.edu.my).

types of traffic generated in the network which are legitimate traffic and attack traffic. A modified version of Agent/Ping with a maximum of 3 retransmissions with 5-second timeouts is used for DNS as implemented in [7]. In our simulation we attach the modified application to the servers. We follow the model set by [8], whereby the request interarrival period is fixed at 10s. The attacker is expected to flood the target name server with excess traffic. The DoS traffic is modeled as constant bit rate (CBR) source. CBR can be generated by the CBR traffic generator in NS-2. We chose different values of delay for applying to the attack start time in order to achieve variability.

III. Materials and methods

This section presents a new attack detection system for DoS against DNS, which uses a machine learning engine to detect and classify attacks. This IDS is a network-node based IDS (NNIDS), which can be implemented on a name server for the purpose of attack detection. Fig. 1 illustrates the overall architecture of our proposed system with input-output data types.

This system starts by gathering packet stream that was received by a name sever. Next, the pre-processor starts analyzing the traffic statistically based on an administrator specified time window of 20s length, which is more than the maximum lookup latency. The parameters that are going to characterize the DNS traffic received by the name server and that constitute the input of the classifier are defined as follows:

1) Throughput of received DNS requests that is defined as the number of received bits at the server. We measured the average value of this metric for the specified time window.

2) Average size of received packets by the server during a monitoring time window.

3) Packet loss that is defined as the number of lost DNS packets that did not reach their destination due to flooding attack traffic.

After preprocessing the traffic and generating the required dataset based on the specified features selection, the machine learning engine is applied. Four different machine learning engines have been evaluated for our system, which three of them are in the category of neural network classifiers and the last one is a modern algorithm based on support vectors. In the following subsections, these engines are introduced in details.

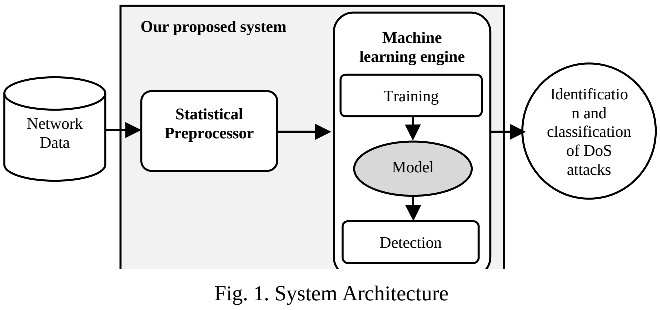

Fig. 1. System Architecture

A. BP neural network

In this paper, we tried to find the optimized BP network that can effectively detect and classify different DoS attacks against DNS. Our BP neural network has three layers. The number of the units in the input layer is equal to the features of input vector which are three features of DNS traffic. There are also three units in the output layer representing different states of normal and DoS attacks: [0 0 0] for normal conditions, [0 0 1] for direct DoS attack and [0 1 0] for the amplification attack. Our main assumptions considered for training process of BP networks are listed as follows: number of epochs = 500, mean squared error (MSE) = 0.00001, training function = Levenberg-Marquardt back propagation (trainlm), activation function = tan-sigmoid. The optimal structure of our network was found by varying the number of hidden neurons from 3 to 13. The best accuracy of the system was for 7 neurons in the hidden layer.

B. RBF neural network

In order to implement an optimized RBF neural network for our classification problem, we need to specify the activation function for the hidden units and the centres and widths of RBFs. The mostly used activation function for the hidden layer is a Gaussian function which has been used for the hidden units in our RBF classifier. The centroid locations have been chosen by K-means clustering algorithm [9], and then the width parameter was calculated using the following equation:

$$\sigma = \frac{\text{maximum distance between any 2 centers}}{\sqrt{\text{number of centers}}} \tag{1}$$

Because of high calculation power requirements, it was not possible to achieve the same MSE as BP neural networks in our preliminary examinations. Therefore, we set the value of MSE to 0.001.

C. SOM neural network

In this experiment, the input vector of three features has been normalized due to the large variations of input values. If the raw data is applied to the network directly, the input samples with higher values may lead to suppress the influence of smaller values. So, the standard normalization given by the following equation was used:

$$nv[i] = v[i] \div \sqrt{\sum_{x} v[x]^2} \tag{2}$$

Different number of neurons was tested to find the best performed network. We obtained sample results by looking at the output of the classifier applied to the trained data and noticed that all normal traffic was clustered between a specified range and the suspicious traffic was outside this cluster indicating a possible attack. When we were confident about the results, the trained network was evaluated by subjecting it to the test data. Therefore, the main assumptions considered for implementing the SOM neural network were as follows: number of epochs = 1000, number of neurons = 25, neighbours topology = Hextop, distance function = Linkdist, ordering phase learning rate = 0.9, ordering phase steps = 1000, tuning phase learning rate = 0.02, and tuning phase neighbour distance = 1.

D. Support vector machines

SVM is another learning and soft computing technique that recently applied to IDSs. The basic SVM algorithm was designed for classification of objects into two classes [7], but many real world problems deal with more than two classes. In our experiments the one-against-all scheme is implemented to overcome this problem. It constructs three binary SVM classifiers, each of which separates one class from all the rest. The *i*th SVM is trained using a training set of positive labels (+1) for *i*th class and negative labels (-1) for all the others. Finally, a sample in our testing data is classified in class, i, which has the maximum value between all three classifiers.

During the training phase, a proper function with the corresponding parameters should be provided. This will be a time consuming process because the machine is trained with different kernel parameters and only the one which is the best performed will be selected for the testing process.

Support vector machines with three radial kernels with gamma = 1.5, 10, and 5, and the optimal regularization parameter C = 100, 1, and 1000000, were used for implementing three classifiers. The Radial basis kernel equation is as follows:

$$K(x,x') = \exp(-gamma \parallel x - x' \parallel^2) \quad (3)$$

IV. Results and discussions

In this section, the performance metrics used to evaluate our proposed system are introduced with their definitions:

- Accuracy, which refers to the proportion of data classified as accurate type in the total data. Accurate situations are True Positive (TP) and True Negative (TN), while false detected situations are False Positive (FP) and False Negative (FN). Accuracy of the system is calculated by the following formula:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \times 100\% \quad (4)$$

- Detection rate (of direct DoS attacks), which refers to the proportion of direct DoS attacks detected among all direct DoS attacks.

- Detection rate (of amplification attacks), similarly refers to the proportion of amplification attacks detected among all amplification attacks. These two metrics are calculated by the following formula:

$$DetectionRate = \frac{TP}{TP + FN} \times 100\% \quad (5)$$

- False Alarm Rate (FAR), which is defined as the percentage of the network traffic that is misclassified by the classifier. It can be calculated using the following formula:

$$FAR = \frac{FP}{FP + TN} \times 100\% \quad (6)$$

Table 1. Performance comparison of different classifiers

Parameter Classifier	DR (direct DoS)	DR (amplification attack)	Accuracy	FAR
BP	99.55	97.82	99	0.28
RBF	99.62	89.48	95.9	0.23
SOM	54.24	65.28	74.40	6.83
SVM	98.26	97	97.6	1.07

Table 1 presents the performance comparison of three neural network classifiers and SVM as well. The results show that a BP neural network outperforms other types of classifiers that have been implemented in this article. It gives us good detection rates for different types of DoS against DNS with an acceptable false alarm rate.

V. Conclusions

This paper has introduced two different types of DoS attacks against DNS which are direct DoS and amplification attacks. The investigation of the impact of DoS attacks against DNS traffic led us to find the suspicious behaviours. Based on these patterns the required traffic data for analytical measurements was simulated using the most flexible network simulator, NS-2. Finally, a machine learning based system is proposed for detecting and classifying DoS attacks against DNS using several traffic statistics. Two different machine learning algorithms were evaluated for the detector engine which are neural network classifiers and support vector machines. The performance comparison results show that a back propagation neural network outperforms other classifiers with 99.55% detection rate for direct DoS attacks, 97.82% detection rate for amplification attacks, 99% accuracy, and 0.28% false alarm rate.

References

[1] D. Davidowicz. (1999). Domain Name System (DNS) Security. Available: http://compsec101.antibozo.net/papers/dnssec/dnssec.html. Accessed, Feb. 2009.

[2] N. Chatzis, "Motivation for behaviour-based DNS security: A taxonomy of DNS-related internet threats," in The International Conference on Emerging Security Information, Systems, and Technologies (SecureWare 2007), Valencia, Spain, Oct., 2007, IEEE Computer Society Press, Washington DC, USA, pp. 36-41.

[3] Y. Wang, M. Hu, B. Li and B. Yan, "Tracking anomalous behaviors of name servers by mining DNS traffic," *LECTURE NOTES IN COMPUTER SCIENCE,* vol. 4331, pp. 351-357, 2006.

[4] R. Vaughn and G. Evron. (Mac. 2006). DNS Amplification Attacks (Preliminary release). Available: http://www.isotf.org/news/DNS-Amplification-Attacks.pdf. Accessed, Nov. 2008.

[5] ICANN. Factsheet root server attack on 6 february 2007. I. C. for Assigned Names and Numbers, pp. 1-6, Mac. 2007.

[6] G. Kambourakis, T. Moschos, D. Geneiatakis and S. Gritzalis, "A fair solution to DNS amplification attacks," in *Second International Workshop on Digital Forensics and Incident Analysis (WDFIA 2007),* Samos, Greece, IEEE Computer Society, Washington, DC, USA, Aug. 2007, pp. 38-47.

[7] J. Mirkovic, A. Hussain, B. Wilson, S. Fahmy, P. Reiher, R. Thomas, W. M. Yao and S. Schwab, "Towards user-centric metrics for denial-of-service measurement," in *Workshop on Experimental Computer Science (Part of ACM FCRC),* San Diego, June, 2007, ACM, pp. 1-14.

[8] K. Lan, A. Hussain and D. Dutta, "Effect of malicious traffic on the network," in *Passive and Active Measurement Workshop (PAM),* San Diego, CA, Apr., 2003.

[9] S. Haykin, *Neural Networks: A Comprehensive Foundation Second Edition (M).* Prentice Hall, 2001.