Distributed Denial of Service (DDoS) Attacks Detection Using Machine Learning Prototype

Manuel S. Hoyos Ll, Gustavo A. Isaza E, Jairo I. Vélez and Luis Castillo O

Abstract The Distributed Denial of Service (*DDoS*) attacks affect the availability of Web services for an indeterminate period of time, flooding the company's servers with fraudulent requests and denying requests from legitimate users, generating economic losses by unavailable rendered services. Therefore, the aim of this paper is to show the process of detection prototype *DDoS* attacks using a supervised learning model by Support Vector Machines (SVM), which captures network traffic, filters HTTP headers, normalizes the data on the basis of the operational variables: rate of false positives, rate of false negatives, rate of classification and then sends the information to corresponding SVM's training and testing sets. The results show that the proposed DDoS SVM prototype has high detection accuracy (99 %) decrease of the false positives and false negatives rates compared to conventional detection models.

Keywords SVM · Machine learning · Intrusion detection · DDoS

1 Introduction

DDoS attacks are a critical issue for companies that have been integrating their technology to public networks, allowing multiple attackers to access data or render services to large companies or countries, such as North Korea, the most recent

M.S. Hoyos Ll(✉) · J.I. Vélez
Department of Computer Science,
University Autónoma of Manizales, Manizales, Colombia
e-mail: shoyos@gmail.com, jvelez@autonoma.edu.co

G.A. Isaza E · L. Castillo O
GITIR Research Group, Department of Systems and Informatics,
University of Caldas, Manizales, Colombia
e-mail: {gustavo.isaza,luis.castillo}@ucaldas.edu.co, lfcastilloos@unal.edu.co

L. Castillo O
Department of Industrial Engineering, National University of Colombia (Manizales),
Manizales, Colombia

S. Omatu et al. (eds.), *DCAI, 13th International Conference*,
Advances in Intelligent Systems and Computing 474,
DOI: 10.1007/978-3-319-40162-1_4

relevant DDoS Attack turning internet communications offline [1]. A DDoS attack consists in to throw tens or hundreds of thousands of requests per second to a server from different locations or IPs; the concept of "Distributor" is concerning that these requests are made from hundreds of thousands of infected machines (commonly called "zombies") which are governed by "botnets" in a coordinated way at the same time, i.e. SYN Flood, Smurf attacks, which are a sum of bandwidth, memory usage and target's processing, usually no servers could handle ending in a collapse of service because it cannot answer every request; therefore it's necessary the development of new techniques and prototypes to detect fraudulent attacks of concurrent requests in an effective and efficient way also it's necessary in order to avoid the unavailability of service and economic losses. Machine learning using SVM have been used with great success in the field of information security and pattern recognition research in different processes of classification, prediction and regression. The application of techniques with a SVM supervised model has large advantages over rule-based techniques, since the generation of the model is based on a statistical model that changes its behavior according to the input parameters defined and based on a training rule that requires human interaction; in the prototype evaluation it was found that the correct classification rate of normal or abnormal requests in the training phase is directly related to standardization and proper selection of the input parameters, allowing the output variables are generated with minimum percentage of misclassification, generating confidence in the generated model and the detection of these behaviors. The paper describes: The contextual reference and some relevant work in section 2. The Section 3 shows the proposed model of the development and application of machine learning prototype. In section 4 some results are evident and in the end, finally in Section 5 the conclusions are presented.

2 Context and Recent Literature

Since 1998 and 1999, DARPA has collected and distributed the first standard dataset for evaluation of intrusion detection systems for computer networks. The first formal, repeatable and statistically significant evaluations of intrusion detection systems are coordinated. These assessments measure probability of detection and false alarm for each system under test and are designed to be simple, to focus on the key issues of technology, and to encourage the broadest possible participation by eliminating problems of security and privacy and provide data types that are commonly used by most systems intrusion detection. The evaluation results suggested that future research should aim to develop new algorithms to detect new attacks but creating static rules or signatures. Since that moment, many experts began working on techniques and models able to resolve this problem. Below are presented some proposed techniques in intrusion detection systems (IDS):

Rules: Responsible for analyzing the traffic that goes through the IDS, classifying the frame as normal or as intrusion. This technique uses a database of knowledge where a set of rules is applied to compare traffic patterns with the parameterized rules in the database [2].

As proposed by [4] on a system based intrusion detection agents with a rules engine based on XML, it can be that the argued model was presented as a distributed intrusion detection system, which consists of three intelligent agents that have specific functions and exchange information via XML sure how SSL and a point-to-point and IAP. The decision to do so and distributed intelligent agents, is based on the intrusion detection systems distributed traditional drawbacks, such as the hierarchy analysis, data refinement, bulky modules at all levels and passive interaction. The distributed sensing system offers more functions intrusions and strong individual actions IDS using intelligent agents.

Neural Networks: Artificial neural networks (ANN) are inspired by the behavior of neurons in the biological world, seeking to emulate in technology. [3]; as proposed by [5], about an automatic defense against distributed denial of service, it can argue model; the authors propose a model of automatic defense that prevents human interaction techniques based on artificial neural networks. An architectural design, which can be easily adapted protocols in each layer of the OSI reference model and algorithms of learning machines. Support Vector Machine (SVM) is a technique based on machine learning, where data is classified by determining a group of support vectors and characteristics to be quantified are described. As proposed by [6, 7], the hybrid system Intrusion Detection (HIDS), based on machine learning and specifically the SVM technique, improve the detection rate. More recent study as [10] presents a better classification using an Artificial Neural Network (ANN) to flag detection engine Known and unknown attacks from genuine traffic.

3 Materials and Method

3.1 Design

In the research field of pattern recognition, machine learning using SVM have been used with great success indifferent classification and regression tools. Some cores used internally by the SVM are: linear, polynomial, radial (RBF) and sigmoid. The design will focus on showing the different layers and levels that have high computational prototype and the components involved in the extraction, filter, standardization, training and evaluation. The prototype design of computational architecture exhibits the following logic, shown in Figure 1.

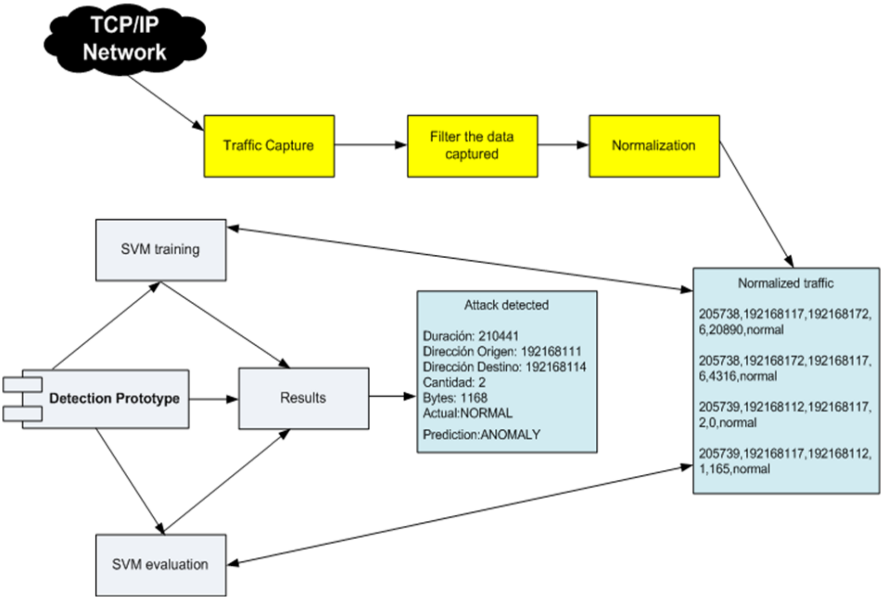


Fig. 1 Logical Architecture Detection Prototype SVM

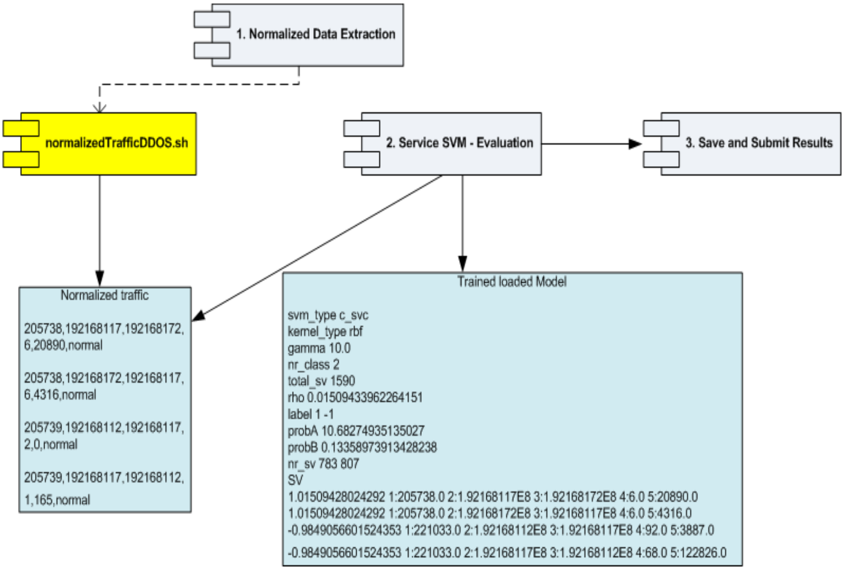
In the Figure 1, the architecture can be seen as the prototype has separate layers capture, filter and standardization of information, with the use of a language to efficiently implement regular expressions required to estimate the volume of information. Evaluation and training will be conducted on a high level language that allows the integration of SVM library and manage the concept of multiprocessing in an effective and efficient way independent of the used operating system.

3.2 Implementation and Training

The detection prototype is technically divided as follows: The collection, filtering and traffic normalization is performed on shell and python scripting, considering the performance of regular expressions and pipes offered by language. Training and evaluation was performed in Java programming language, given that the SVM library was installed on it and the features mentioned before were in the attacks generator. Data collection, filtering and Normalization aims to recover the network traffic, filter and normalizes the data, taking into account the operational variables defined in the design stage, TCPDUMP analyzer was used on the server to capture network traffic in the *Dataset Collection*, after a filtering and normalization stages are applied. The training phase is to receive standardized information and send it to the SVM to perform statistical clustering process of abnormal and normal requests in the generated model.in a first step the whole process of capturing network traffic will take place in the second stage filter HTTP headers on the basis of the information downloaded, in the third stage the standardization process where operational variables are selected to be used, as fourth stage integration process and training prototype with a standard percentage of traffic, and as a last step the respective evaluation of the prototype in the classification of anomalous or normal entry.

The prototype integrates *libsvm*[1] library in JAVA, which allows training, evaluating and generating a statistical model, based on standardized information. The internal structure of the generated model can be seen in Figure 2 that corresponds to a matrix groups of the entered operational variables and generates real value qualifying results: (Normal: 1.0 and Abnormal: -0.0).

The model is physically stored on a server path and is based on the evaluation phase or detection. The evaluation phase is to receive the standard traffic and send it to the SVM to perform the process of detecting the attack based on the model that was generated in the training stage. Below diagram components of the training and evaluation process are shown in Figure 2. The evaluation process has as its starting point the traffic sent by users and it’s standardized at the initial stage, the prototype loads the generated model and discusses each record individually, in order to compare the statistical model generated in step training and identify indeed if it is generating an attack.


1. Normalized Data Extraction
normalizedTrafficDDOS.sh
2. Service SVM - Evaluation
3. Save and Submit Results
Normalized traffic
Trained loaded Model

```
205738,192168117,192168172,
6,20890,normal

205738,192168172,192168117,
6,4316,normal

205739,192168112,192168117,
2,0,normal

205739,192168117,192168112,
1,165,normal
```

```
svm_type c_svc
kernel_type rbf
gamma 10.0
nr_class 2
total_sv 1590
rho 0.01509433962264151
label 1 -1
probA 10.68274935135027
probB 0.13358973913428238
nr_sv 783 807
SV
1.01509428024292 1:205738.0 2:1.92168117E8 3:1.92168172E8 4:6.0 5:20890.0
1.01509428024292 1:205738.0 2:1.92168172E8 3:1.92168117E8 4:6.0 5:4316.0
-0.9849056601524353 1:221033.0 2:1.92168112E8 3:1.92168117E8 4:92.0 5:3887.0
-0.9849056601524353 1:221033.0 2:1.92168117E8 3:1.92168112E8 4:68.0 5:122826.0
```

Fig. 2 Components in the evaluation phase or detection

4 Results

The final standard file structure displayed by the operational variables defined in the design stage, presented in Table 1.

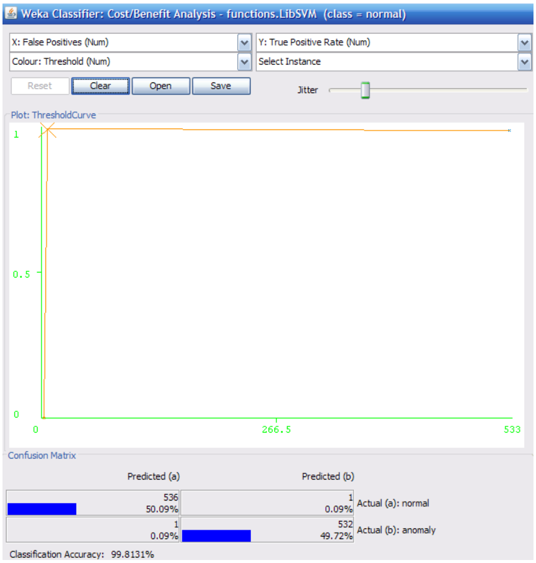
[1] https://www.csie.ntu.edu.tw/~cjlin/libsvm/

Table 1 Structure of normalized attributes

Variable	Description	Type
DURATION	Time in seconds for each request.	Continuous
IP_SRC	IP Source address	Continuous
IP_DEST	IP destination address.	Continuous
IP_SRC_COUNT	Number of requests from the source IP.	Continuous
BYTES	Number of bytes sent from a source to a destination.	Continuous
CLASS	Traffic Classification (Normal, Anomalous).	Discrete

For the information gathering phase, were sent a set of normal and abnormal requests for parallel and sequential process, with attacks prototype generator system test data, sending five iterations of 500 requests over a period determined time of 15 minutes. Total pooled and stored records 2698 between normal and anomalous traffic were normalized. The complete data normalized traffic (normal 1349, anomalous 1349), are divided into the training stage, randomly selecting 60% of the dataset (normal 809 abnormal 809), the other 40% (normal 539 abnormal 539) for evaluation. With the percentage defined training and evaluation, proceed to perform the respective tests the prototype, in order to determine if it has an acceptable level of detection. Before looking at the results of the training and testing phase, it has been used the metrics applied in [8, 9].

It was randomly selected 60% of the dataset (normal 809, 809 anomalous), and training to perform the respective prototype, saving the information into a file and classifying the data manually with normal and anomalous traffic. This process must be done manually because the SVM is a supervised method which requires prior training for model generation and respective evaluation of the attack. After training, 40% of the dataset of machine learning (normal 539, 539 anomalous) is evaluated, based on the model generated in the previous stage and analyzing the percentage of right and wrong classification of the instances. To make the respective evaluation, the information is saved in a file and the data is automatically sorted, placing as the default normal traffic. Performance metrics shown in the evaluation phase have the same relevance in the training phase because they allow us to see whether or not the SVM is correctly classifying the requests. In this case, the evaluation is a good rating level. ROC curves based viewing rate of false positives and true positives (detection rate) are shown in Figure 3. In the same way as in the training phase in the X axis is the false positive and in the Y axis is the true positive, if the value on the Y axis is close to 1 and the value on the X axis approaches to 0, it means that the events will be better able to detect and therefore greater ability to discriminate between normal and anomalous behavior. In order to make the comparison between a technique that uses a rules engine as SNORT, against another technique using SVM learning machines were taken on the data collected for the evaluation (normal 539 anomalous 539) and became the detection process. The prototype implemented the SVM technique had a significant improvement



Weka Classifier: Cost/Benefit Analysis - functions.LibSVM (class = normal)

X: False Positives (Num)

Y: True Positive Rate (Num)

Colour: Threshold (Num)

Select Instance

Confusion Matrix

	Predicted (a)	Predicted (b)	
	536 50.09%	1 0.09%	Actual (a): normal
	1 0.09%	532 49.72%	Actual (b): anomaly

Classification Accuracy: 99.8131%

Fig. 3 ROC curve

(approximately 10%) in the classification accuracy when compared with the intrusion detection system SNORT using the rules defined by human experts.

The internal statistical model of SVM identified anomalous or normal data in the evaluation phase without the need to add or modify the rules as another IPS (Intrusion Prevention Systems), causing the intrusion detection process is carried out quickly, automatically and without human interaction. The results obtained comparing metrics using a conventional Open Source IPS [11] is presented in Table 2.

Table 2 SVM DDoS Prototype Performance Results metrics vs Conventional IDS

	SNORT **Rules Engine IDS**		**DDoS Prototype** **using SVM**	
	Anomalous	Normal	Anomolous	Normal
Anomalous	TVP: 88.9%	TFP: 12.2%	TVP: 98.9%	TFP: 0.2%
Normal	TFN	TVN	TFN	TVN
Accuracy: 89%	Sensibility: 88%		Accuracy: 99%	Sensibility: 98%

5 Conclusions

The training process and tuning the Machine Learning from the standard data set are the basis for the generated model and it has an acceptable percentage of classification at the time of the evaluation of the prototype in a production environment with real information. The selection of metrics in the intrusion detection problem: false positive rate, false negative rate, rate classification, ROC curves, allow having a standard of comparison against other models. The application of techniques with supervised training as SVM model, has large advantages over the technique based on rules, since the generation of the model is based on a statistical model that changes its behavior according to the input parameters defined in the training and based on rules it requires human interaction. In the prototype evaluation was found a better classification rate for normal and anomalous requests in the training phase, is directly related to standardization and proper selection of input parameters, allowing output variables to be generated with minimum percentage of misclassification, generating reliability in the generated model and the detection of these behaviors.

References

1. Keizer, G.: Garden-variety DDoS attack knocks North Korea off the Internet. Recovered March 13, 2015 (2014). http://www.computerworld.com/article/2862652/garden-variety-ddos-attack-knocks-north-korea-off-the-internet.html
2. Chan, A., Ng, W., Yeung, D., Tsang, E.C.: Refinement of rule-based intrusion detection system for denial of service attacks by support vector machine. In: Proceedings of 2004 International Conference on Machine Learning and Cybernetics, vol. 7, pp. 4252–4256 (2004)
3. Kartalopoulos, S.: Understanding Neural Networks and Fuzzy Logic: Basic Concepts and Applications, 1st edn. Wiley-IEEE Press (1996)
4. Liu, W.-T.: Research on intrusion detection rules based on XML in distributed IDS. In: International Conference on Machine Learning and Cybernetics, vol. 3, pp. 1400–1403, 12 de 07 de 2008
5. Mukkamala, S., Sung, A.: Detecting denial of service attacks using support vector machines. In: The 12th IEEE International Conference on Fuzzy Systems, FUZZ 2003, vol. 2, pp. 1231–1236 (2003)
6. Seufert, S., O' Brien, D.: Machine Learning for Automatic Defence Against Distributed Denial of Service Attacks. In: IEEE International Conference on Communications, ICC 2007, pp. 1217–1222, 24–28 de Junio de 2007
7. Subbulakshmi, T., Shalinie, S., GanapathiSubramanian, V., BalaKrishnan, K., AnandKumar, D., Kannathal, K.: Detection of DDoS attacks using Enhanced Support Vector Machines with real time generated dataset. In: 2011 Third International Conference on Advanced Computing (ICoAC), pp. 17–22, 14.16 de Diciembre de 2011
8. Isaza, G.A., Castillo, L.F., Trujillo, M.L., Marulanda, C.E.: Modelo híbrido de neuroclasificación y clustering en el problema de detección de intrusiones. Vector, 69–77 (2012)

9. Isaza, G.A., Castillo, A., Lopez, M.F., Castillo, L.: Towards Ontology-based intelligent model for Intrusion Detection and Prevention. Journal of Information Assurance and Security **5**(2), 376 (2010)
10. Saied, A., Overill, R.E., Radzik, T.: Artificial Neural Networks in the Detection of Known and Unknown DDoS Attacks: Proof-of-Concept. In: Communications in Computer and Information Science, vol. 430, pp. 300–320. Springer-Verlag, Heidelberg (2014). doi:10.1007/978-3-319-07767-3_28
11. Kacha, C., Shevade, K.A.: Comparison of Different Intrusion Detection and Prevention Systems. Intl. Journal of Emerging Technology and Advanced. Engineering **2**(12), 243–245 (2012)