دار المنظومة
DAR ALMANDUMAH
الرواد في قواعد المعلومات العربية

العنوان:	An Intrusion Detection System for DOS Attacks Based on Neural Networks
المصدر:	مجلة العلوم الاقتصادية والسياسية
الناشر:	الجامعة الأسمرية الإسلامية - كلية الاقتصاد والتجارة
المؤلف الرئيسي:	Bentaher, Omran Ali
مؤلفين آخرين:	Al Bhbah, Atia M.(Co-Auth)
المجلد/العدد:	ع7
محكمة:	نعم
التاريخ الميلادي:	2016
الشهر:	يونيو
الصفحات:	430 - 445
رقم MD:	765769
نوع المحتوى:	بحوث ومقالات
اللغة:	English
قواعد المعلومات:	EcoLink
مواضيع:	الشبكات العصبية، أمن الشبكات، هجوم حجب الخدمة، نظام كشف التسلل
رابط:	http://search.mandumah.com/Record/765769

© 2021 دار المنظومة. جميع الحقوق محفوظة.
هذه المادة متاحة بناء على الإتفاق الموقع مع أصحاب حقوق النشر، علما أن جميع حقوق النشر محفوظة. يمكنك تحميل أو طباعة هذه المادة للاستخدام الشخصي فقط، ويمنع النسخ أو التحويل أو النشر عبر أي وسيلة (مثل مواقع الانترنت أو البريد الالكتروني) دون تصريح خطي من أصحاب حقوق النشر أو دار المنظومة.

An intrusion detection system for DoS attacks based on neural networks

Omran Ali Bentaher

Atia M. Albhbah

***Abstract*:**

Intrusion detection systems (IDSs) have become an essential component of computer security to detect attacks that occur despite the best preventive measures. A problem with majority of current intrusion detection systems is their rule-based nature. In this paper, we propose an optimized neural network based IDS for detecting DoS attacks. The proposed system consists of Multiple Layered Perceptron (MLP) decision block and a feature reduction preprocessing subsystem. The system was optimized and tested on benchmark KDDCUP' 99 dataset. Several experiments have been conducted to choose the important features from full set of 41, based on three factors: training time, testing time and detection accuracy. Final optimized MLP IDS provides superior accuracy of 98.5%, substantially better than other referential IDS systems published up to now.

***Keywords*:** Denial-of-Service attack, Feature selection, Intrusion Detection Systems, Neural Networks.

Introduction:

Denial of service (DoS) is a type of attack in which an attacker issues a huge amount of packets to congeal specific servers' services, consequently blocking legitimate users from normal access to the services [1], [2]. Many methods have been developed to secure the network infrastructure and communication over the Internet, among them the use of firewalls, encryption, and virtual private networks. Intrusion detection systems (IDS) are a relatively new addition to such techniques. There are two different ways of classifying an IDS, the first way is to classify based on the method of detection, in the form of either signature detection or anomaly detection. In signature detection known representation of intrusions are stored in the IDS and then compared to system activity. When a known intrusion matches an aspect of system use, an alert is raised to the IDS analyst. Known representations of intrusion are termed signatures. Signatures must be created to exactly match the characteristics of a specific intrusion and no other activity to avert false positives. On the other hand, an anomaly detection based IDS detects intrusions by searching for abnormal network traffic. The anomaly detection IDS gathers a set of data from the system activity of the user. This baseline dataset is then deemed "normal use." If the user deviates from the normal use pattern, an alarm is raised. One of the most commonly used approaches is rule based intrusion detection systems [3]. Unfortunately, expert systems require frequent rule updates to remain efficient. This design approach usually results in an inflexible detection system that is unable to detect an attack if the sequence of events is even slightly different from the predefined profile.

While misuse based detection is generally favored in commercial products due to it is predictability and high accuracy, in academic research anomaly detection is typically conceived as a more powerful method due to it theoretical potential for addressing novel attacks. Conducting a thorough analysis of the recent research trend in anomaly detection, one will encounter several machine learning methods, including neural networks, reported to have a very high detection rate of 98% while keeping the false alarm rate at 1% [4]. A key aspect of any anomaly detection technique is the nature of the input data. In order to have benchmark data for training and testing such type of IDS, Lincoln Labs at MIT sponsored by the U.S. KDDCUP' 99 collected real data sets which become the widely used data set for the evaluation of IDS [5]. More information about this data will be given in section II.

The main goal of our study is to show that carefully designed IDS based on feature reduction technics, followed by a MLP classifier has superior performance over existing IDS reported in available literature. The organization of paper is as follows. In section II we describe the KDDCUP' 99 benchmark data sets, commonly used in IDS community. In section III we expose so called empirical feature ranking algorithm, along with selection of the best MLP architecture. In section IV we give experimental results, comparison to other approaches, and asses final accuracy of proposed MLP based IDS.

KDDCUP' 99 data set description:

Since 1999, KDDCUP' 99 has been the most wildly used data set for the evaluation of anomaly detection methods within IDS

community. This data set is prepared by Stolfo et al. [6] and is built based on the data captured in DARPA' 98 IDS evaluation program [7]. DARPA' 98 is about 4 gigabytes of compressed raw (binary) tcpdump data of 7 weeks of network traffic, which can be processed into about 5 million connection records, each with about 100 bytes. The two weeks of test data have around 2 million connection records. KDDCUP' 99 training dataset consists of approximately 4,900,000 single connection vectors each of which contains 41 features and is labeled as either normal or an attack, with exactly one specific attack type. The simulated attacks fall in one of the following four categories: Denial of Service Attack (DoS), User to Root Attack (U2R), Remote to Local Attack (R2L), Probing Attack.

KDDCUP' 99 features can be classified into three groups: Basic features, Traffic features, which comprises "Same host" and "Same service" features, and Content features.

Feature reduction and MLP architecture selection:

Feature selection and ranking is an important issue in intrusion detection. Of the large number of features that can be monitored for intrusion detection purpose, which are truly useful, which are less significant, and which may be useless? The question is relevant because the elimination of useless features (the so-called audit trail reduction) enhances the accuracy of detection while speeding up the computation, thus improving the overall performance of an IDS. In our cases where there are no useless features, by concentrating on the most important ones we may well improve the time performance of an

IDS without affecting the accuracy of detection in statistically significant ways.

The feature ranking and selection problem for intrusion detection is similar in nature to various engineering problems that are characterized by:

Having a large number of input variables x = (x 1, x 2, ..., xn) of varying degrees of importance to the output y; i.e., some elements of x are essential, some are less important, some of them may not be mutually independent, and some may be useless or irrelevant (in determining the value of y)

Lacking an analytical model that provides the basis for a mathematical formula that precisely describes the input-output relationship, y = F (x)

Having available a finite set of experimental data, based on which a model (e.g. neural networks) can be built for simulation and prediction purposes

Due to the lack of an analytical model, one can only seek to determine the relative importance of the input variables through empirical methods. A complete analysis would require examination of all possibilities, e.g., taking two variables at a time to analyze their dependence or correlation, then taking three at a time, etc. This, however, is both infeasible (requiring numerous experiments) and not infallible (since the available data may be of poor quality in sampling the whole input space). Therefore, the technique of deleting one feature at a time to rank the input features and identify the most

important ones for intrusion detection is applied. One input feature is deleted from the data at a time; the resultant data set is then used for the training and testing of the classifier. Then the classifier's performance is compared to that of the original classifier (based on all features) in terms of relevant performance criteria. The importance of the feature is ranked according to a set of rules based on the classifier comparison. To perform feature ranking, the classifier is trained with the 41-feature set. After that for each feature is done the following procedure:

- Delete the feature from the (training and testing) data.
- Use the 40-feature data set to train the classifier.
- Analyze the performance of the classifier using the test set.
- Rank the importance of the feature according to the rules.

To rank the importance of the feature, three main parameters were considered: accuracy, training time and testing time. Each feature is ranked according to the rules below that are applied to the result of comparison of the original 41-feature classifier and the 40-feature classifier. For example, if one attribute is necessary (which means it is very important for detection) then removing it from feature vectors would result in decreasing accuracy and increasing at least one time (be it training or testing), which results in overall performance degradation. However, if attribute is not needed for detection, it's removal should result in performance improvement, be it higher or unchanged detection rate and processing time decreased. Formally we can explicitly formulate these rules:

Attribute is necessary for detection if (after its removal):

- Accuracy decreases and either training and/or testing time increases.
- Accuracy remains unchanged, but both training and testing time increase.

Attribute is ranked not needed for DoS detection if (after it's removal):

- Accuracy is unchanged or increases, but both training and testing time decrease.

In other situations, attribute is ranked as important but not necessary for detection.

- Accuracy remains unchanged, but both training and testing time increase.

Attribute is ranked not needed for DoS detection if (after it's removal):

- Accuracy is unchanged or increases, but both training and testing time decrease.

In other situations attribute is ranked as important but not necessary for detection.

According to our experiments and rules of empirical-based reduction, after 41 experiments performed, a set of necessary, important and not needed attributes for DoS category detection have been extracted.

TABLE 1: SELECTION OF MLP NEURAL NETWORK ARCHITECTURE

Iteration	Number of neurons hidden ayers	RMSE
1	20, 10, 5	0.195
2	14, 9, 3	0.149
3	14, 9, 4	0.015
4	14, 9	0.002
5	14, 10	0.010

As a classifier we choose MLP neural network, due to its ability to restore arbitrary mappings. A feed forward neural net is composed of a number of consecutive layers, each one connected to the next by a synapse/connection. MLP architecture consists of one input, two hidden and one output layer. The input layer consists of 41 neurons because the KDDCUP' 99 data set contains 41 features for a TCP/IP packet to be used for attack detection. The output layer consists of two neurons that classify normal packets from abnormal packets. There is no certain mathematical approach for obtaining the optimum number of hidden layers and number of their neurons [6]. For choosing optimum set of hidden layers and its number of neurons, a comparison is made for many architectures and optimum is selected based on smallest root means square error, as shown in the Table 1.

After choosing optimal neural networks architecture, and application of the described feature ranking procedure, we obtain relevance of all attributes, as it is shown in Table 2.

TABLE 2: FEATURE RANKS

#	Feature name	Rank
1	Duration	Necessary
2	Protocol type	Important, but not necessary
3	Service	Necessary
4	Flag	Not needed for DoS detection
5	Source bytes	Necessary
6	Destination bytes	Necessary
7	Land	Important, but not necessary
8	Wrong fragments	Necessary
9	Urgent	Important, but not necessary

#	Feature name	Rank
10	Hot	Important, but not necessary
11	Failed logins	Important, but not necessary
12	Logged in	Not needed for DoS detection
13	# Compromised	Not needed for DoS detection
14	Root shell	Important, but not necessary
15	Su attempted	Not needed for DoS detection
16	# Root	Not needed for DoS detection
17	# File creations	Important, but not necessary
18	# Shells	Not needed for DoS detection
19	# Access files	Necessary
#	Feature name	Rank
20	# Outbound cmds	Important, but not necessary
21	Is hot login	Not needed for DoS detection
22	Is guest login	Important, but not necessary
23	Count	Necessary
24	Srv count	Necessary
25	Serror rate	Necessary
26	Srv serror rate	Necessary
27	Rerror rate	Necessary
28	Srv rerror rate	Necessary
29	Same srv rate	Important, but not necessary
30	Diff srv rate	Important, but not necessary
31	Srv diff host rate	Not needed for DoS detection
32	Dst host count	Necessary
33	Dst host srv count	Necessary
34	Dst host same srv rate	Important, but not necessary
35	Dst host diff srv rate	Necessary
36	Dst host same src port rate	Necessary
37	Dst host srv diff host rate	Important, but not necessary
38	Dst host serror rate	Necessary
39	Dst host srv serror rate	Necessary
40	Dst host serror rate	Necessary
41	Dst host srv serror rate	Necessary

Experimental results:

In this section, we show testing results of MLP trained with the full feature set, necessary feature set and union of necessary and important (but not necessary) feature sets. Testing has been performed on classifiers trained both with the complete training set and 10% data set. Results, regarding overall accuracy, training and testing time compared to 41 feature full training set are shown in the Table 3. Times needed for training and testing with reduced feature sets are given compared to times needed

TABLE 3: RESULT OF TESTING MLP IDS FOR DOS ATTACK

Set of features	Accuracy	Training time	Testing time
SCENARIO 1: full training set			
full 41 feature	95.2%	1	1
necessary + important	97.2%	0.71	1.04
necessary feature set	98.5%	0.74	0.72
SCENARIO 2: 10% training set			
full 41 feature set	93.1%	1	1
necessary + important	95.0%	0.79	1.12
necessary feature set	96.7%	0.80	0.79

To perform training and testing with the full featured sets. From the Table 3 follows that the best classifier is classifier trained with the full training set using only necessary features. This classifier has best overall accuracy, provides the quickest decision and gets trained as second-best.

After the training process was complete, testing was conducted basically in two steps. In the first step system was tested against the training dataset, in order to examine how well neural networks 'learned' the training dataset after the training process. In the second step of the testing, trained neural networks were tested against a dataset, which is not a part of the training set, in order to examine generalization performance of the trained networks. In both testing steps performance of the neural networks was evaluated by examining the number of false positive and false negative decisions. Hence classifier is tested with the novel DoS attacks available only in the testing set: apache 2, mail-bomb, process table and udpstorm. Results

of testing these attacks with classifier trained with a full training set with only the necessary features are given in Table 4.

TABLE 4: DOS ATTACK DETECTION PERFORMANCE

Attack	Accuracy	FP	FN
Apache 2	98%	1%	1%
mail-bomb	99%	0%	1%
Process table	100%	0%	0%
udpstorm	97%	1%	2%

The best classifier provides almost 100% accuracy with no false positives or negatives for the process table attacks, while making 1% false positives and false negatives for apache 2, 1% false negatives for mail-bomb and 1% false positives and 2% false negatives for udpstorm attacks.

TABLE 5: COMPARISON OF PROPOSED MLP IDS AND SELECTED PUBLISHED SYSTEMS [11]

Feature set	Accuracy	Training time	Testing time
SCENARIO 1: full training set			
empirical based necessary feature set	98.5%	0.74	0.72
SVDF reduced feature set	92.1%	0.66	0.61
LGP reduced feature set	93.2%	0.67	0.59
MARS reduced feature set	91.6%	0.73	0.71
SCENARIO 2: 10% training set			
empirical based necessary feature set	96.7%	0.80	0.79
SVDF reduced feature set	88.7%	0.72	0.68
LGP reduced feature set	89.4%	0.71	0.63
MARS reduced feature set	85.1%	0.75	0.70

On previously known attacks, this classifier provides almost 100% accuracy. It can be concluded that classifier trained with

proposed set of features is a very good neural network solution for detection of DoS attacks.

To prove the worthiness of proposed MLP IDS for DoS detection, its results are compared to other IDS reported in the literature, trained on various KDDCUP' 99 reduced feature sets, see Table 5. There are 4 reduced sets of features with which classifiers are trained and tested:

1- MLP IDS with empirical-based feature reduction set containing only necessary features (11 features),

2- Support Vector Decision function (SVDF) based feature reduction (5 features) [11],

3- Linear Genetic Programing (LGP) based feature reduction (5 features) [11],

4- Multivariate Adaptive Regression Splines (MARS) based feature reduction (5 features) [11].

Although the proposed MLP model itself is not the fastest, if compared to models trained by reduced sets found in literature, it provides the highest detection rate.

Conclusion:

We are witnessed flourishing of machine learning approaches to general area of Cyber Security [12], [13]. In our views this phenomena comes from very idea of machine learning paradigm: to adapt our decision according to available data, always having in mind that final criterion will be applied on new unseen data. If we carefully

follow this principle during design of an IDS, then obtained results can be compared to the best alternative approaches. Our investigation presented in this paper supports this very general statement.

REFERENCES

1- R.K.C. Chang, "Defending against Flooding-Based Distributed denial-of-Service Attack: A Tutorial," IEEE *Communication Magazine*, October 2002, 42-51.

2- W.Y. Luo, "A Lightweight System of Detecting DoS/Probe Attacks Based on Packet Header", National *Taiwan University of Science and Technology.*

3- D. E. Denning, "An intrusion detection model," *IEEE Transactions on Software Engineering*, vol. 13, no. 2, pp. 222–232, 1987.

4- M. Shyu, S. Chen, K. Sarinnapakorn, and L. Chang, "A novel anomaly detection scheme based on principal component classifier," *Proceedings of Third IEEE International Conference on Data Mining (ICDM03)*, pp. 172–179, 2003.

5- KDD Cup 1999. Available on:

http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html, Ocotber 2007.

6- J. McHugh, "Testing intrusion detection systems: a critique of the 1998 and 1999 DARPA intrusion detection system evaluations as performed by Lincoln laboratory," *ACM Transactions on Information and System Security, vol. 3, no. 4,* pp. 262–294, 2000.

7- Fox, Kevin L., Henning, Rhonda R., and Reed, Jonathan H. (1990). "A Neural Network Approach Towards Intrusion Detection". *In Proceedings of the 13th National Computer Security Conference.*

8- Uwe Aickelin, Julie Greensmith, Jamie Twycross, "Immune System Approaches to Intrusion Detection–A Review" *Natural computing, Springer Netherlands, Volume 6, Number 4 /* December, 2007, pp 413-466

9- http://www.mathworks.com/matlabcentral. [Accessed: 15 Jan 2016].

10- Martin Riedmiller, Rprop - Description and Implementation Details *Technical Report, Institute fur Logik, University of Karlsruhe,* January 1994.

11- Vemuri, V. (ed.), Enhancing Computer Security with Smart Technology, CRC Press, 2005.

12- Sumeet Dua, Xian Du, Data Mining and Machine Learning in Cybersecurity, CRC Press, 2011.

13- Mehadu Masud, Latifur Khan and Bhavani Thuraisingham, Data Mining Tools for Malware Detection, CRC Press, 2012.