Training And Testing Anomaly-Based Neural Network Intrusion Detection Systems

Loye Lynn Ray

Cyber Security and Information Assurance Department, Adjunct Associate Professor, University of Maryland University College, 3501 University Blvd East, Adelphi, MD

Address: Tel: +011 717 718 5727, e-mail: loye.ray@faculty.umuc.edu

Abstract- Networks are up against detecting dynamic and unknown threats. Anomaly-based neural network (NN) intrusion detection systems (IDSs) can manage this if trained and tested accordingly. This requires the IDS to be evaluated on how well it can detect these intrusions. Evaluating NN IDSs can be a complex and difficult task. One needs to be able to measure the convergence rate and performance (detection and failure) rate of the IDS. This paper explores the different methods used by researchers to train and test their IDS models. It also found that the data used can effect the results of training and testing the NN IDS models.

Keywords- intrusion detection; neural network; KDD 99; convergence rate; performance rate.

1. Introduction

Today attacks are more dynamic and sophisticated that can easily break into computer networks. These can produce intrusions that can change from one day to the next. Network intrusion detection systems (IDSs) must be able to adapt quickly to identify these threats. Garuba, Liu and Fraites [10] saw this and agreed that it is becoming more difficult to protect networks with an increasing number of new and unknown threats. Neural network based IDSs can help by using anomaly detection methods looking at abnormal behavior in a network system. These are composed of three layers: input, hidden and output layers. Each is composed of neurons for processing the information. To help in processing this information are different kinds of algorithms such as back propagation and genetic algorithms. Anomaly-based Neural Networks (NN) IDSs can adapt to these new threats if trained and tested effectively. This involves measuring the performance and convergence rates of the NN IDS in detecting these threats. However, the majority of IDSs today face the challenge of coping with low detection and high convergence rates that can misidentify traffic [17]. Complex attack patterns and similarities to normal traffic can cause these problems. This paper compares the training and testing techniques used by researchers to produce effective intrusion detection of this new attack environment.

2. Description

Configuring a NN IDS to handle these new threats involves two parts: training and testing phases. It is important for all networks to be properly trained to recognize intrusions and be able to detect them in a live computer environment.

2.1. Data

Each phase requires having traffic data available to perform these two phases. Researchers commonly use a simulated dataset called Knowledge Discovery Dataset 1999 (KDD 99) to

train and test neural networks. This dataset was devised in 1999 by the federal government and based on simulating US Air Force local area network traffic. The KDD 99 dataset is composed of a full training set, 10% training set and testing set [6]. Researchers may choose one of these three datasets. This dataset is composed of normal and attack categories Denial of Service (DoS), Probe, Remote-to-User (R2U) and User-to-Root (U2R). The dataset is public domain and thus one of the reasons it is widely used. However, a few researchers have devised ways of creating or collecting real traffic data. Real traffic can be collected from real network traffic by monitoring a server or network device. Others generate network traffic by using automated security tools to simulate attacks against a computer network. Either one is used to train and test NN IDSs for use in a network. A percentage of the dataset is set aside for creating a training set and test set. Each set is composed of about ten percent of the total dataset. This is because of the large size of the dataset. The sample size used can affect the convergence and performance rates of the NN IDS.

2.2. Training

Training a NN IDS involves using a dataset (simulated or live) to prepare the network to recognize various normal and abnormal traffic behavior. The KDD99 dataset is the most used for evaluating IDSs according to Engen, Vincent and Phalp [6]. This is because no other one is publicly available. The amount of time it takes to train a network is called the convergence rate. This is measured in the number of times the network training sequence is repeated measured in epochs. Ideally, one wants to have a low convergence rate that shows it takes little time to training the network to detect individual intrusions based on a very small percentage of error. To train the network, one establishes an error value and introduces the training dataset to the NN IDS. After one cycle through the dataset, the percentage of errors are measured. The process repeats until the percentage of errors drops below the threshold set. This establishes the convergence rate. Also the performance rate is initially established based on the training dataset. The training may also stop early if the error rate begins to increase [6].

2.3. Testing

Testing the NN IDS is similar to the training process. After training, the NN IDS is ready for testing using a test dataset. This dataset is smaller than the training dataset to ensure that the network can detect intrusions it was trained to detect. Also the test dataset is ran through once to determine the performance rate. This rate is composed of a separate detection rate and failure rate. The detection rate is how well the network correctly identifies the traffic as normal or intrusion. The failure rate is the percentage of traffic misidentified. A lack of measuring the failure rate could allow attackers to get through if the data rate is high [10].

3. Discussion

Training and testing NN IDS models can be affected by the type of data used. The most commonly used are the publicly available KDD 99 dataset and data collected from real or simulated networks. Various research models were examined on how different datasets effected the training and testing of their models. The amount or type of data used seem to vary between models that affected the convergence and performance rates of the NN IDS.

3.1. Using KDD 99 dataset

Ahmad, Abdullah and Alghamdi [2] utilized the KDD 99 dataset for both training and testing phases of their NN IDS model. Their reason was that real traffic data was not available and too complicated to obtain. The model was designed, trained and tested for detecting denial of service (DoS) attacks. Thus, they only used the DoS portion of the KDD 99 dataset. The training phase was completed within 1000 epochs [2]. The testing phase consisted of two parts. The first part used the same training data to verify the training. Different data was used in the second part to measure the ability of the IDS to generalize how well it detects intrusions. The model showed a 96% detection accuracy and 0-1% false positive error rate for six types of attacks under the DoS category [2].

Gong, Fu and Cai [11] also used the KDD 99 dataset but reduced the number of attack dimensions to train and test their model. They used only half the number of attack dimensions in the 10 percent dataset from the KDD 99 dataset for both training and testing. Their model showed a detection accuracy of between 52.5 and 100 percent depending on the attack being detected. According to Gong, Fu and Cai [11], it improved IDS performance but also reduced the amount of input data needed.

Shum and Malki [19] also used the KDD 99 dataset but trained and tested their model using different datasets. They utilized very small (200 and less) elements in their model. This adjustment of the KDD 99 dataset helped them achieve 100% accuracy in training and detecting attack patterns. Shum and Malki [19] confined their results to detection rates for training, normal traffic, known and unknown attacks. These are very broad attack types when one is determining what kind of response to do to combat this threat.

Kandeeban and Rajesh [17] used all four categories of the KDD 99 dataset in their training and testing. They utilized 50K patterns for the training dataset and 10K patterns for the testing dataset. However, they only detected DoS attacks. The model was trained over 5000 epochs. The detection rate of their model showed a detection rate between 11.4 to 99.2% for training while testing showed a range of 92 to 96%. However. No values were given for convergence or failure rates.

Hoque, Mukit and Bikas [13] only used the KDD 99 dataset to train and test their model. They used the 10 percent dataset (nearly 500K records) for training and a corrected dataset (over 300K records) for testing. All four categories of the KDD 99 dataset were used. The authors used a confusion matrix to depict the training and testing results. The training results were 6% (U2R), 50% (R2L), 54% (probe), 77% (normal and 92% (DoS). The testing results were slightly worst and better consisting of 70% (normal), 71% (probe), 99% (DoS), 19% (U2R) and 5% (R2L). No explanation was given in the differences in detection rates. Also no failure rates were measured to compare against other researcher findings.

Haddadi, Khanchi, Shetabi and Derhami [12] took the KDD 99 dataset and broke it into two datasets composed of different number of records in each. Training and test data from each attack group was broken down with 80% used for training and the rest for testing. The reason for this was to reduce the chances that the IDS would remember threats. The researchers checked for normal, DoS, probe, R2L and U2R attack categories used in the KDD 99 dataset. The first dataset was about 5K records while the second dataset was 10K records. The first dataset took just less than 600 epochs to train while the second dataset took over 610 epochs. The detection rates ranged from 35 to 99%. There was little difference between the two data set detection rates so less data can be used to provide an adequate detection rate. However, no failure rates were measured.

Farid, Darmont, Harbi, Hoa and Rahman [8] used 10% of the KDD 99 dataset but reduced the number of attributes to suit their research. There was an imbalance in the number of training samples to testing where more samples were used in testing for probe, U2R and R2U checks. They tested using both the full 41 attributes of the KDD 99 dataset and a reduced 19 attribute. The detection rates were around 99% with a different between each value of attributes of less than 0.2%. This is not much difference. However, the false positive rate differed from 0.01 to over 1 % between the testing using 41 versus 19 attributes. This makes the use of 19 attributes more appealing because of the lower false positive rate.

Al-Sharafat and Naoum [3] also used the KDD 99 dataset but set up their model to classify different classes of attacks based on significance. The 10% training dataset from the KDD 99 dataset was used for training the IDS. The testing dataset from the KDD 99 dataset was also used. The detection rate for their model ranged from 29 to

97% depending on the attack category. However, no false positive rate was measured.

3.2. Using real or simulated data

Wang and Ma [20] generated their own dataset using automated tools such as netpoke and Hgod. They also followed a similar training and testing phase for checking their model. However, unlike the others, Wang and Ma [20] tested against no specific type of attacks. Their best training detection rate was 88.4% and training took 200 seconds. This required 2000 epochs to achieve these values. Testing their model resulted in a 92% detection rate that was better than the value during the training phase.

Jing-xin, Zhi-ying and Kui [15] used real traffic data collected using a SNORT IDS. They collected, for each kind of intrusion, 80 samples for training and 70 for testing. Training took nearly 14 hours and over 100K epochs to complete. Testing looked at known and unknown attack types. The known attack types were detected between 97 to 100% with a false error rate of 0-1.5%. The unknown attack types included backdoor, DoS and FTP. Its detection rate was 86 to 96% with an error rate of 3 to 13%. These findings showed that selecting certain features to detect is important to IDSs according to Jing-xin, Zhi-ying and Kui [15].

Hua and Xiaofeng [14] used WinPcap to collect data for training and testing their model. The data collected was broken up into training (700) and testing (1200). Training of the model was completed after about 27K epochs. The detection rate achieved was 93% while the error rate was 7%.

Rastegari, Saripan and Rasid [18] created their model to detect DoS attacks in Domain Name Service (DNS) systems. Their model required specific data to be able to train and test their model. They used a OTcl program to simulate these attacks. This is because no simulated dataset was available for DoS attacks against DNS. The results of their training and testing took about 11 seconds for training and the detection rate was 99%. The false alarm rate was just under 0.3%. These showed a good value for measuring the efficiency of an IDS.

4. Analysis

The characteristics of the input data is important to any NN IDS. They define the parameters for training, testing accuracy and how the system will perform [1]. These provide the basis of evaluating the effectiveness of NN IDSs. Also there is a link to how well we train an IDS and the features selected. These can positively affect the performance and accuracy of the NN IDS [1]. Carefully picking the features can affect the convergence rate and time for training the NN IDS. Too many features and the convergence rate and training time will increase. Abdulla, Al-Dabagh and Zakaria [1] recommended that these features be fixed to ensure optimum results.

There seems to be a trend that researchers are using the KDD 99 dataset because of its public availability. However, they may be neglecting the idea that any model devised, trained and tested using this dataset will not be effective in today's networks.

4.1. Limitation of input data

The KDD 99 dataset was made in 1999 and found to be outdated for today's threats. Today's NN IDSs need to be able to train and test against more current simulated traffic patterns in order to effectively detect intrusions. Otherwise attackers could get through and cause damage to a network. Engen, Vincent and Phalp [7] found that the KDD 99 dataset also didn't contain a balanced amount of data for each of the threat categories. They found that the U2L and R2L composed less than 1% of the training data. Also, this dataset hasn't been updated and doesn't contain real traffic data. The KDD 99 dataset was found not to be representative of today's new and unknown traffic patterns [7]. Even though this dataset continues to be used by researchers, its usefulness could be questioned when used to train and test today's NN IDSs. A new dataset is needed that contains updated attack

patterns for NN IDSs designers to use. Real traffic data is sometimes collected by researchers in different ways. However, these are generated by using automated tools and restricted to a limited number and type of intrusions. This limits the effectiveness of using the NN IDS in a real network that is exposed to many new and dynamic intrusions. The result is no standard method for collecting real traffic data to form an improved dataset for training and testing NN IDSs. Also duplicate data was found in the KDD 99 dataset. This can have a negative impact to training and skew the results from testing the NN IDS [7]. They can also increase the convergence rate and reduce the detection accuracy [8]. The result of using a dataset with duplicates can also invalidate results and complicate interpretation of intrusions.

4.2. Convergence Rate

There were several things found about the researchers training their NN IDS models. An important fact was that few of the researchers measured the convergence rate of their model. This involves setting the error threshold and establishing an epoch value for training their model. The amount of times (epoch) it took to train their models ranged from 600 to 100K. The best time came from the Haddadi, Khanchi, Shetabi and Derhami [12] model at around 600 epochs. However, they didn't provide how long it took them to accomplish the 600 epochs. Rastegari, Saripan and Rasid [18] took only 11 seconds to training but failed to mention how many epochs it took. Wang and Ma [20] provided the best complete described convergence rate of 200 seconds after 2000 epochs.

4.3. Limitations of training

The amount of input data can negatively affect the convergence and performance rates of the NN IDS. Redundant and irrelevant data can complicate the IDS model and reduce the accuracy [8]. It was found that too much input data could reduce the convergence rate thus causing the network too long in training. Bo, Zhang and Cheng [5] discovered this in their research of back propagation neural network IDSs. Also too little input data could reduce the detection rate and increase error rate. It was also found that researchers didn't always use the standard ten percent dataset from the KDD 99 dataset. Instead they constructed their own and used various sample sizes. It was also found that the researchers manipulated the data size in order to get good readings for their study. Engen, Vincent and Phalp [7] support these claims by finding that researchers would use the KDD 99 dataset training and test data, only the training set or a smaller subset of the dataset. Bo, Zhang and Cheng [5] found and removed some redundancy within the KDD 99 dataset before they used the dataset. This could have negative effects on the convergence rates and training times for NN IDSs if left in. Haddadi, Khanchi, Shetabi and Derhami [12] support this claim and found that less data is better because it reduces the computational overhead.

4.4. Performance Rate

The performance rate is composed of a detection rate and failure rate. This is the main measure of the effectiveness of a NN IDS to detect intrusions and attacks to a network. The best values for detection and failure rates from the models reviewed, Rastegari, Saripan and Rasid [18] has the best detection rate of 99% and a error rate of less than 0.3%. However, their model was restricted to detecting only DoS attacks that limits its effectiveness in detecting dynamic types of different intrusions. The Ahmad, Bdullah and Alghamdi [2] also had a good detection rate and low false error rate. However, they too devised a model for only DoS attack detection. The best overall for different types of attacks was from the Fraid, Darmont, Harbi, Hoa and Rahman [8] model which had a 99% detection rate and an error rate of around 1%.

4.5. Limitations of testing

Testing most NN IDSs use the common KDD 99 dataset. However, there isn't standard dataset that used real traffic data to overcome the outdated information in the KDD 99 dataset. Few researchers were found using some form of real traffic data. These had to generate their own data

for specific types of attacks. Balram and Wiscy [4] created their own data to use in detecting scanning attacks. Restricting the testing to a small number of intrusions weakened the effectiveness of the NN IDSs. Also researchers used a small sample dataset to control the amount of errors in testing. A good example was when Gao and Tain [9] used a varying training sample size between 60 to 120 and testing sample size of 20 to 50 samples. This reduced the effectiveness of the network because normal networks receive large amounts of data traffic to detect intrusions from. These small sample sizes were found to produce questionable results. Some had perfect detection rates and low error rates. One had perfect scores but only used 200 data samples to reach these results. Another limit found in testing was that some researchers didn't measure a complete performance rate. They only measured the detection or failure rates. This also negatively affects the operation of IDS in detecting abnormal traffic where poor performance can allow attackers to get through. According to Garuba, Liu and Fraites [10], this is a possibility when failure rates are not part of the measurement. Joo, Hong and Han [16] also noticed that most researchers concentrated on the detection accuracy of the IDS instead of also considering the failure rates.

5. Conclusion

This study explored the training and testing of anomaly-based NN IDSs. It was found that NN IDSs need to be trained using a detailed dataset based on current threats. However, the common KDD 99 dataset is too old and there is a lack of a standard dataset containing real traffic data to train the network. A new dataset is needed that has no duplicates and combines both training and testing datasets [7]. Al-Sharafat and Naoum [3] also supports this claim and recommends a reduction of the amount of features in the dataset. However, this doesn't address the need for current traffic data. Further work is needed in developing a simple dataset composed of both simulated and real traffic. Also the amount and type of input data was found to be important to properly train and test a network. Too much or too little input sample data can negatively affect the convergence and performance rates. More research is necessary to determine the optimum amount of data to use for detecting today's intrusions. In the meantime, researchers have recommended that the KDD 99 dataset be used to look at challenges to generic machine learning and not evaluate IDSs [7].

Today's networks must be able to detect dynamic and unknown attacks to a network according to Garuba, Liu and Fraites [10]. To accomplish this requires a NN IDS to have a low convergence rate (training), high detection rate and low failure rate (errors). The study shows that further work is needed to determine a balance of these three factors to optimize a NN IDS for use in a live network. Failure to consider this can degrade the efficiency of the IDS [16].

References

[1] S. M. Abdulla, N. B. Al-Dabagh, and O. Zakaria, "Identify features and parameters to devise an accurate intrusion detection system using artificial neural network", *World Academy of Science, Engineering and Technology*, Vol. 70, pp. 627-631, November 2010.

[2] I. Ahmad, A. B. Abdullah and A. S. Alghamdi, "Application of artificial neural network in detection of DOS attacks", *Second International Conference on Security of Information and Networks*, North Cyrus, Turkey, pp. 229-234, 6-10 October 2009.

[3] W. S. Al-Sharafat and R. Naoum, "Development of genetic-based machine learning for network intrusion detection (GBML-NID)", *World Academy of Science, Engineering and Technology*, Vol. 55, pp. 20-24, July 2009.

[4] S. Balram and M. Wiscy, "Detection of TCP SYN scanning using packet counts and neural network", *Fourth International Conference on Signal Image Technology and Internet Based Systems*, Bali, Indonesia, pp. 646-649, 30 November–3 December 2008.

[5] J. Bi, K. Zhang and X. Cheng, "A new method of data processing in the intrusion detection system with neural network", *Second International Workshop on Education Technology and Computer Science*, Wuhan, Hubei, China, pp. 343-345, 6-10 March 2010.

[6] V. Engen, J. Vincent, and K. Phalp, "Exploring discrepancies in findings obtained with the KDD Cup '99 data set", *Intelligent Data Analysis*, Vol.15, pp. 251-276, April 2011.

[7] V. Engen, J. Vincent and K. Phalp, "Enhancing network based intrusion detection for imbalanced data", *International Journal of Knowledge-based and Intelligent Engineering Systems*, Vol. 12, Iss. 5/6, pp. 357-367, December 2008.

[8] D. M. Farid, J. Darmont, N. Harbi, N. H. Hoa and M. Z. Rahman, "Adaptive network intrusion detection learning: Attribute selection and classification", *World Academy of Science, Engineering and Technology*, Vol. 60, pp. 154-158, December 2009.

[9] M. Gao and J. Tian, "Network intrusion detection method based on improved simulated annealing neural network", *2009 International Conference on Measuring Technology and Mechatronics Automation*, Zhangjiajie, Hunan, China, pp. 261-264, 11-12 April 2009.

[10] M. Garuba, C. Liu and D. Fraites, "Intrusion techniques: Comparative study of network intrusion detection systems", *Fifth International Conference on Information Technology: New Generations*, Las Vegas, NV, pp. 592-598, 7-9 April 2008.

[11] W. Gong, W., Fu and L. Cai, "A neural network based intrusion data fusion model", *Third International Joint Conference on Computational Science and Optimization*, Huangshan, Auhui, China, pp. 410-414, 28-31 May 2010.

[12] F. Haddai, S. Khanchi, M. Shetabi and V. Derhami, "Intrusion detection and attack classification using feed-forward neural network", Second International Conference on Computer and Network Technology, Bangkok, Thailand, pp. 262-266, 23-25 April 2010.

[13] M. S. Hoque, M. A. Mukit and M. A. N. Bikas, "An implementation of intrusion detection system using genetic algorithm", *International Journal of Network Security & Its Applications*, Vol. 4, No.2, pp. 109-119, March 2012.

[14] J. Hua and Z. Xiaofeng, "Study on the network intrusion detection model based on genetic neural network", *2008 International Workshop on Modeling, Simulation and Optimization*, Hong Kong, China, pp. 60-64, 27-28 December 2008.

[15] W. Jing-xin, W. Zhi-ying and D. Kui, "A network intrusion detection system based on the artificial neural networks", *Third International Conference on Information Security*, Pudong, Shanghai, China, pp. 166-170, 14-16 November 2004.

[16] D. Joo, T. Hong and I. Han, "The neural network models for IDS based on the asymmetric costs of false negative errors and false positive errors". *Expert Systems With Applications*, Vol. 25, pp. 69-75, July 2003.

[17] S. S. Kadeeban and R. S. Rajesh, "A genetic algorithm based elucidation for improving intrusion detection through condensed feature set by KDD99 dataset", *Information and Knowledge Management*, Vol. 9, No. 1, pp. 1-9, December 2011.

[18] S. Rastegari, M. I. Saripan, and M. F. A. Rasid, "Detection of denial of service attacks against domain name system using neural networks", *International Journal of Computer Science Issues*, Vol. 6, No. 1, pp. 23-27, November 2009.

[19] J. Shum and H. A. Malki, "Network intrusion detection system using neural networks", *Fourth International Conference on Natural Computation*, Jinan, China, pp. 242-246, 18-20 October 2008.

[20] H. Wang and R. Ma, "Optimization of neural networks for network intrusion detection", *First International Workshop on Education Technology and Computer Science*, Wuhan, Hubei, China, pp. 418-420, 7-8 March 2009.