Feature Selection for Effective Anomaly-Based Intrusion Detection

Neveen I. Ghali
Faculty of Science, Al-Azhar University - Egypt

Summary
Intrusion Detection system (IDs) has become the main research focus in the area of information security. Most of the existing IDs use all the features in the network packet to evaluate and look for known intrusive patterns. Some of these features are irrelevant and redundant. The drawback of this approach is a lengthy detection process and degrading performance of an ID system. In this paper a new hybrid algorithm RSNNA (Rough Set Neural Network Algorithm) is used to significantly reduce a number of computer resources, both memory and CPU time, required to detect an attack. The algorithm uses Rough Set theory in order to select out feature reducts and a trained artificial neural network to identify any kind of new attaches. Tests and comparison are done on KDD-99 data set used for The Third International Knowledge Discovery and Data Mining Tools Competition, which was held in conjunction with KDD-99 The Fifth International Conference on Knowledge Discovery and Data Mining, The results showed that the proposed model gives better and robust representation of data as it was able to select features resulting in a 83% data reduction and 85%-90%time reduction and approximately 90%reduction in error in detecting new attacks.

Key words:
Rough Set theory, feature selection, intrusion detection, dimensionality reduction

1. Introduction

Intrusion detection is defined to be the process of monitoring the events occurring in a computer system or network and noticeably different from normal system activities and thus detectable [12].

An Intrusion Detection system (IDs) is a program that analyzes what happened or what has happened during an execution and tries to find indications that the computer has been misused. An IDs does not eliminate the use of preventive mechanism but it works as the last defensive mechanism in securing the system [9].

Based on processing of data to detect attacks, IDs can also be classified into two types: misuse-based systems and anomaly-based systems. While the former keeps the signatures of known attacks in the database and compares new instances with the stored signatures to find attacks, the latter learns the normal behavior of the monitored system and then looks out for any deviation in it for signs of intrusions. It is clear that misuse based IDs cannot detect new attacks and we have to add manually any new attack signature in the list of known patterns. IDs based on anomaly detection, on the other hand, are capable of detecting new attacks as any attack is assumed to be different from normal activity. However anomaly based IDs sometimes sets false alarms because it cannot differentiate properly between deviations due to authentic user's activity and that of an intruder [10].

This paper aims to make anomaly-based intrusion detection feasible. Many approaches have been proposed which include statistical [2], machine learning [13], data mining [3], and immunological inspired techniques [6]. The work of Zhang et al. [15], exploited the capability of rough set theory in coming up with the classification rules in determining the categories of attacks in IDs. Their findings showed that rough set classification attained high detection accuracy and the feature ranking was fast. Unfortunately they did not mention the features used for the classification process. Rawat et al. [10] applied the rough set theory in extracting decision rules for ID on BSM audit files for the DARPA'98. Godinez et al. [4] reduced the features of only 8 BSM log files for the DARPA'98 using rough set theory resulting in 66% reduction in the number of attributes, the output reduct is validated using association pattern.

In this work we aim to filter out redundant, superfluous information, and significantly reduce a number of computer resources, both memory and CPU time, required to detect an attack. We work under the consideration that intrusion detection is approached at the level of execution of operating system calls, rather than network traffic. So our input data is noiseless and less subject to encrypted attacks. In our reduction experiments, we used the data set [5] used for The Third International Knowledge Discovery and Data Mining Tools Competition, which was held in conjunction with KDD-99 The Fifth International Conference on Knowledge Discovery and Data Mining. This data is considered a standard benchmark for intrusion detection evaluations.

Manuscript received January 2, 2009
Manuscript revised March 20, 2009

This paper suggests rough set theory as a reduction tool and feed forward neural networks as a learning tool for the developed system, first to test the efficiency of the system after the removal of superfluous features and then to efficiently detect any intrusions.

The paper is organized as follows, in the followed section we give a declaration of what is meant by an IDs, Section three gives a brief introduction to Rough Set theory especially the parts used in the problem of dimension reduction. Section four explains the proposed algorithm. In section five experimental setup and results are presented. Finally in section six conclusion and future work are shown.

2. Intrusion Detection System Model

During a certain intrusion, a hacker follows fixed steps to achieve his intention, first sets up a connection between a source IP address to a target IP, and sends data to attack the target. Generally, there are four categories of attacks as given in [1, 6, 14], they are:

(i Dos: Denial of Service. This kind of attack consumes a lot of computing, memory resources and denying the legitimate requests. The means of achieving this are varied from buffer overflows to flooding the systems resources. For example: ping-of-death, teardrop, smurf, and SYN flood.

(ii U2R: User to Root. Unauthorized access to super user root privilege. This kind of attack starts out with normal user accessing the system and gradually exploiting system vulnerabilities to gain super user access. For example buffer overflow attacks.

(iii Probe: Attacker scans the network to gather information about the network and find the system's known vulnerabilities. These vulnerabilities will be exploited to attack the system. For example port scanning.

(iv R2L: Remote to Local. Unauthorized access from a remote to local machine. An attacker who does not have an account exploits some systems' vulnerabilities to gain local access. For example guessing password.

For each TCP/IP connection, 41 various quantitative and qualitative features were extracted plus 1 class label.

Table 1: Network Data Feature Label

Label	*Network Data Features*	*Label*	*Network Data Features*	*Label*	*Network Data Features*
A	Duration	O	Su_attempted	AC	Same_srv_rate
B	Protocol_type	P	Num_root	AD	Diff_srv_rate
C	Service	Q	Num_file_creations	AE	Srv_diff_host_rate
D	Flag	R	Num_shells	AF	Dst_host_count
E	Sec_byte	S	Num_access_files	AG	Dst_host_srv_count
F	Dst_byte	T	Num_cutbounds_cmds	AH	Dst_host_same_srv_rate
G	Land	U	Is_host_login	AI	Dst_host_diff_srv_rate
H	Wrong_fragment	V	Is_guest_login	AJ	Dst_host_same_src_port_rate
I	Urgent	W	Count	AK	Dst_host_srv_diff_host_rate
J	Hot	X	Sev_count	AL	Dst_host_server_rate
K	Num_failed_login	Y	Serror_rate	AM	Dst_host_srv_serror_rate
L	Logged_in	Z	Sev_serror_rate	AN	Dst_host_rerror_rate
M	Num_comprised	AA	Rerror_rate	AO	Dst_host_srv_rerror_rate
N	Root_shell	BB	Srv_rerror_rate		

Table 1 shows all the features found in a connection. For easier referencing, each feature is assigned a label (A to AO). Some of these features are derived features. These features are either nominal or numeric.

Log files are naturally represented as a table, a two dimensional array, where rows stand for objects (in our case system calls) and columns for their features. These tables may be unnecessarily redundant [4].

IDs must therefore reduce the amount of data to be processed. This is very important if real-time detection is desired. The easiest way to do this is by doing an intelligent input feature selection. Certain features may contain false correlations, which hinder the process of detecting intrusions. Further, some features may be redundant since the information they add is contained in other features. Extra features can increase computation time, and can impact the accuracy of IDs [9].

Rough Set theory is a mathematical tool for approximate reasoning. A reduct is a minimal subset of features with the same capability of objects classification as a whole set of features. Reduct computation of rough set corresponds to feature ranking for IDs [14]. Below is a brief overview of Rough Set theory and how reducts are obtained.

3. Rough Set Theory Preliminaries

In production environments, output data are often vague, incomplete, inconsistent, and of a great variety, getting in the way of its sound analysis. Fortunately, the theory of Rough Sets has been specially designed to handle these kinds of scenarios. In Rough Sets every object of interest is associated with a piece of knowledge indicating relative membership. This knowledge is used to derive data classification and is the key issue of any reasoning, learning, and decision making [8].
Knowledge, acquired from human or machine experience, is represented as a set of examples describing attributes of two types, condition and decision [11].
Rough Set theory deals with inconsistencies, uncertainty and incompleteness by imposing an upper and a lower approximation to set membership. It has been successfully used as a selection tool to discover data dependencies and find out all possible feature subsets and remove superfluous information. Hence, a reduct is a minimal subset of attributes with the same capability of objects classification as the whole set of attributes [8, 14].
The following definitions as given in [8] shows the reduct derivation.

Definition 1:
Knowledge is represented by means of a table called an Information System given by $S = <U,A,V,f>$; where $U = \{x_1, x_2, \ldots, x_n\}$ is a finite set of objects of the universe (n is the number of objects); A is a non empty finite set of features, $A=\{a_1, a_2, \ldots, a_m\}$; $V= \cup_{a \in A} V_a$ and V_a is a domain of feature a; $f: U \times A \rightarrow A$ is a total function such that $f(x,a) \in V_a$ for each $a \in A$, $x \in U$. If the features in A can be divided into condition set C and decision feature set D; i.e. $A=C \cup D$ and $C \cap D = \Phi$. The information system A is called decision system or decision table.

Definition 2:
Every $B \subseteq A$ yields an equivalence relation up to indiscernibility, $IND_A(B) \subseteq (U \times U)$, given by: $IND_A(B) = \{(x,x') : \forall a \in B \ a(x) = (x')\}$ a reduct of A is the least $B \subseteq A$ that is equivalent to A up to indiscernibility. i.e., $IND_A(B) = IND_A(A)$.

4. Proposed Algorithm

In this paper a new hybrid algorithm RSNNA (Rough Set Neural Network algorithm) is proposed to filter out redundant, superfluous information required to detect an attack. The algorithm uses Rough Set theory in order to filter out superfluous, redundant information and a trained artificial neural network to identify any kind of new attaches. The main steps can be summarized as follows

- Discover data dependencies and deduce the features reduct using Rough Set theory.
- Build a feed forward back propagation neural network using MATLAB used for training and classifying new system calls for the reduced features.

5. Experimental Setup and Results

We ran our experiments on a system with a 1.7GHz Pentium IV processor and 512 MB DDR RAM running Windows XP. All the processing was done using MATLAB®. MATLAB's Neural Network Toolbox was used for designing a feed forward back propagation neural network, whereas rough set operations were done in ROSETTA. It is a toolkit developed by Ohrn [7] used for data analysis using Rough Set theory.

The algorithms supplied by ROSETTA library supports two types of discernibility: i) *Full*: in this case the reducts are extracted relative to the system as a whole, ii) *Objects*: This kind of discernibility extract reducts relative to a single object. We are interested in Genetic Algorithm, which is a reduct extraction algorithms supplied by ROSETTA library, It is used to find minimal hitting sets and it gives less number of reducts as compared to Johnson's algorithm [4,7,12].

A MATLAB feed forward neural network program has been developed for training process. The network has to discriminate the different kinds of anomaly-based intrusions. 6385 sets of input samples (6128 sets for training and 257 sets for testing) with 41 features, representing 10% of KDD'99 dataset with 2.1MB size.

After enough experimentation, it was inferred that one hidden layer with 8 neurons and one output was giving the

optimum results for classification of all data before feature selection.

By applying the reduction algorithm defined by Rough Set theory, using ROSETTA for the 6385 samples. The resulting reduced features are only 7 given in table 2. This gives 83% reduction in input data.

Table 2: Significant features induced by Rough Set theory

Label	*Network data feature*
E	Sec_byte
F	Dst_byte
W	Count
X	Sev_count
AF	Dst_host_count
AG	Dst_host_srv_count
AJ	Dst_host_same_src_port_rate

Feed forward back propagation neural network architecture with one hidden layer and 4 neurons was giving the optimum results for the 7 reduced featured data.

Table 3 shows the optimal experimental results for running the 2 designed neural networks with varying weights. The table shows results of comparing the classification for the input data before and after feature selection according to the training time, measured in minutes:seconds and mean squared error (mse) for the training process.

Table 3: Experimental results before and after feature selection

	Before feature selection (41 features)			*After feature selection (7 features)*		
	train	**test**	**mse**	**Train**	**test**	**Mse**
Trial 1	1:25	0:30	0.6	0:08	0:02	0.02
Trial 2	1:08	0:20	0.3	0:10	0:01	0.04

The results in table 3 show that the proposed model gives better and robust representation of data as it was able to reduce the number of attributes resulting in a 83% data reduction and 85%-90%time reduction and approximately 90% reduction in error in detecting new attacks.

6. Conclusion

This paper addresses the problem of dimensionality reduction using features selection. The proposed algorithm RSNNA (Rough Set Neural Network Algorithm) uses Rough Set theory in order to select out feature reducts and a trained artificial neural network to identify any kind of new attaches. Tests and comparison are done on KDD-99 data set used for The Third International Knowledge Discovery and Data Mining Tools Competition, which was held in conjunction with KDD-99 The Fifth International Conference on Knowledge Discovery and Data Mining. The test data contains 4 kinds of different attacks in addition to normal system call.

The experimental results for the selected input system calls show that the proposed model gives better and robust representation of data as it was able to process all kinds of intrusions simultaneously. And to reduce the number of attributes resulting in 83% reduction in input data and 85%-90% time reduction and approximately 90% reduction in mean squared error in detecting new attacks. Meantime it significantly reduce a number of computer resources, both memory and CPU time, required to detect an attack. This shows that the proposed algorithm is very reliable in intrusion detection.

References:

[1] Adetunmbi, A., Falaki, S., Adewale O., and Alese, B., *Network Intrusion Detection Based On Rough Set And K-Nearest Neighbor*, International Journal of Computing and ICT Research, Vol. 2 No. 1, June 2008

[2] Bace, R., Mell, P., *Intrusion Selection systems*, NIST special publication on intrusion detection system, 2001.

[3] Denning, D., *An Intrusion-Detection Model*, IEEE computer society symposium on research in security and privacy, pp. 118-131, 1986.

[4] Godinez, F., Hutter, D., Monroy R., *Attribute Reduction for Effective Intrusion Detection*, AWIC 2004, LNAI 3034, 2004.

[5] Hettich, S. and Bay, S. D., The UCI KDD Archive [http://kdd.ics.uci.edu]. Irvine, CA: University of California, Department of Information and Computer Science. 1999.

[6] Lane, T., *Machine Learning Techniques for the Computer Security*, PhD thesis, Purdue University, 2000.

[7] Ohrn, A., Komorowski, J., *A Rough Set Toolkit for Analysis of Data*, In Proceedings of the third Joint conference on Information Sciences, Vol(3), pp.403-407, USA, 1997 available on http://www.idi.ntnu.no/~aleks/rosetta

[8] Pawlak, Z., *Rough Sets: Theoretical Aspects of Reasoning about Data*, Kluwer Academic Publishers, 1991

[9] Ramos, V.., Abraham, A., *ANTIDS: Self-Organized Ant-Based Clustering Model for Intrusion Detection System*, 4th IEEE International Workshop on Soft Computing, vol.29, pp. 977-986, 2005.
[10] Rawat, S., Gulati V., Pujari A., *A Fast Host-based Intrusion Detection System Using Rough Set Theory*, Transaction on Rough Sets IV, LNCS 3700, 2005.
[11] Sousa, T., Silva, A., Neves, A., *Particle Swarm Based Data Mining Algorithms for Task Classification*, Parallel Computing 30, pp.767-783, 2004.
[12] Srinoy, S., Kurutach, W., Chimphlee, W., Chimphilee, S., *Network Anomaly Detection Using Soft Computing*, Proceedings of World Academy of Science, Engineering and Technology, vol. 9, pp.140-144, 2005.
[13] Sundaram, A., *An Introduction to Intrusion Detection*, Crossroads: The ACM student magazine, 2(4), 1996.
[14] Zainal, A., Maarof M., Shamsuddin S., *Feature Selection using Rough-DPSO in Anomaly Intrusion Detection*, ICCSA 2007, LNCS 4705, 2007.
[15] Zhang, L., Zhang G., Yu, L., Bai, Y., *Intrusion Detection Using Rough Set Classification*, Journal of Zhejiang University Science, Vol. 5(9), 2004.