Volume : 1, Issue : 1
January - June 2011

ISSN : 2229 - 3515

international journal of ADVANCES IN SOFT COMPUTING TECHNOLOGY

Editor-in-Chief
Dr.Vaka Murali Mohan

Published by
BHAVANA RESEARCH CENTER

An Immunity-Based Technique to Characterize Intrusions in Computer Networks

Suryaprakash Reddy, T[1]*and Harinath Reddy, B[2]

1.Dept. of CSE, Krishna Chaitanya Institute of Technology & Sciences, Markapuram

2. Dept. of CSE, TRR Engineering College, Inole (V), Patancheru (m), Medak (Dt), AP

KeyWords:
Artificial immune system,
biological systems modeling,
detector generation,
genetic algorithms,
intrusion detection.

Abstract: *This paper presents a technique inspired by the negative selection mechanism of the immune system that can detect foreign patterns in the complement (nonself) space. In particular, the novel pattern detectors (in the complement space) are evolved using a genetic search, which could differentiate varying degrees of abnormality in network traffic. The paper demonstrates the usefulness of such a technique to detect a wide variety of intrusive activities on networked computers. We also used a positive characterization method based on a nearest-neighbor classification. Experiments are performed using intrusion detection data sets and tested for validation. Some results are reported along with analysis and concluding remarks.*

1. Introduction:

The security of networked computers plays a strategic role in modern computer systems. This task is so complicated because the determination of normal and abnormal behaviors in computer networks is hard, as the boundaries cannot be well defined. One of the difficulties in such a prediction process is the generation of false alarms in many anomaly based intrusion detection systems. However, fuzzy logic is an important solution to reduce the false alarm rate in determining intrusive activities. Network intrusion detection is the problem of detecting unauthorized use of, or access to, computer systems over a network. Two broad approaches exist to tackle this problem: anomaly detection and misuse detection. An anomaly detection system is trained only on examples of normal connections, and thus has the potential to detect novel attacks. However, many anomaly detection systems simply report the anomalous activity, rather than analyzing it further in order to report higher-level information that is of more use to a security officer.

On the other hand, misuse detection systems recognize known attack patterns, thereby allowing them to provide more detailed information about an intrusion. However, such systems cannot detect novel attacks.

*** Dr. B. Harinath Reddy**
Professor, *Dept. of CSE ,*
TRR Engineering College
Inole (V), Patancheru (M),Medak (Dt), AP
Ph. No.: 91-9676771450
E-mail: haribhoomireddy@yahoo.com

Artificial immune systems (AISs) are biologically inspired problem solvers that have been used successfully as intrusion detection systems (IDSs). The biological immune system is an autonomic system for self-protection, which has evolved over millions of years probably through extensive redesigning, testing, tuning and optimization process. The powerful information processing capabilities of the immune system, such as feature extraction, pattern recognition, learning, memory, and its distributive nature provide rich metaphors for its artificial counterpart.

Intrusion detection based upon computational intelligence is currently attracting considerable interest from the research community. Characteristics of computational intelligence (CI) systems, such as adaptation, fault tolerance, high computational speed and error resilience in the face of noisy information fit the requirements of building a good intrusion detection model. The research contributions in each field are systematically summarized and compared, allowing us to clearly define existing research challenges, and to highlight promising new research directions. The findings of this review should provide useful insights into the current IDS literature and be a good source for anyone who is interested in the application of CI approaches to IDSs or related fields by many of the experts such as Mohammad Saniee Abadeh et al [1] proposed a parallel genetic local search algorithm (PAGELS) to generate fuzzy rules capable of detecting intrusive behaviors in computer networks. Dasgupta, D et al [2] described security agent architecture, called CIDS, which is useful as an administrative tool for intrusion detection.

Specifically, it is an agent-based monitoring and detection system, which is developed to detect malfunctions, faults, abnormalities, misuse, deviations, intrusions, and provide recommendations in the form of common intrusion detection language. Shelly Xiaonan Wu and Wolfgang Banzhaf [3] presented an overview of the research progress in applying CI methods to the problem of intrusion detection. The scope of this review will encompass core methods of CI, including artificial neural networks, fuzzy systems, evolutionary computation, artificial immune systems, swarm intelligence, and soft computing. Franciszek Seredynski and Pascal Bouvry [5] presented an architecture of an anomaly detection system based on the paradigm of artificial immune systems (AISs). Simon T. Powers and Jun He [6] presented a hybrid system with the aim of combining the advantages of both approaches. Specifically, anomalous network connections are initially detected using an artificial immune system. Dipankar Dasgupta [7] focused on building an autonomic defense system, using some immunological metaphors for information gathering, analyzing, decision making and launching threat and attack responses. Azzedine Boukerche et al [8] proposed a security system for fraud detection of intruders and improper use of both computer system and mobile telecommunication operations. This technique is based upon data analysis inspired by the natural immune human system. Gerry Dozier et al [9] compared a genetic hacker with 12 evolutionary hackers based on particle swarm optimization (PSO) that have been effectively used as vulnerability analyzers (red teams) for AIS-based IDSs.

The problem of characterizing the normal and abnormal behavior of a system in network environment is very complex. The general assumption is that the normal behavior of a system can often be characterized by a series of observations over time. Also, normal system behavior generally exhibits stable patterns when observed over a period of time. There are multiple approaches to such anomaly detection and most of them work by building a model or profile of the system that reflects its normal behavior. A simple approach is to define thresholds (upper and lower) for each monitored parameter of the system and, if a parameter exceeds this range, it is considered an abnormality. The most common approach uses a statistical model to calculate the probability of occurrence of a given value; the lower the probability, the higher the possibility of an anomaly. In general, statistical approaches model individually different variables that represent the state of the system. This approach, however, ignores two important facts.

1) Normalcy depends on time: A value that might be considered normal at a given time might be abnormal at a different time. In general, we must discuss normal (or abnormal) temporal patterns instead of normal (or abnormal) values.

2) The notion of normalcy depends on correlations among different parameters: The independent values of two different parameters might be considered normal, but their combination might show abnormality or otherwise.

Other approaches also build models to predict the future behavior of systems or processes based on the present and past states.

This paper proposes an approach that does not rely on structured representation of the data and uses only positive data to build a normal profile of the system. It is applied to perform anomaly detection for network security, but it is a general approach that can be applied to different anomaly detection problems.

First, a positive characterization (PC) technique is presented. It is applied to different data sets and the results are analyzed. Later, a negative characterization (NC) technique is proposed that alleviates the efficiency issue of the PC technique. This technique is inspired by artificial immune systems ideas and it attempts to extend Forrest's (selfnonself). Two-class approach to a multiclass approach. Specifically, the nonself space will be further classified in multiple subclasses to determine the level of abnormality. Experiments are performed and the results are compared with the ones produced by the PC technique.

A. Anomaly Detection Problem Definition

The purpose of anomaly detection is to identify which states of a system are normal and which are abnormal. The states of a system can be represented by a set of features.

Definition 1System State Space:

A state of the system is represented by a vector of features xi = (xi1… xin)∈[0.0,1.0]n . The space of states is represented by the set S is subset of [0.0, 1.0] n. It includes the feature vectors corresponding to all possible states of the system.

The features can represent current and past values of system variables. The actual values of the variables could be scaled or normalized to fit a defined range [0.0, 1.0].

Definition 2- Normal Subspace (Crisp Characterization):

A set of feature vectors Self is subset of S represents the normal states of the system. Its complement is called Nonself and is defined as NonSelf=S-Self. In many cases, we will define the Nonself (or Nonself) set using its characteristic function Xself:[0.0,1.0] -> {0,1}.

Xself(x)={1, if x ∈ Self 0, if x ∈ Nonself}

The terms self and nonself are motivated by the natural immune system. In general, there is no sharp distinction between the normal and abnormal states; instead, there is a degree of normalcy (or, conversely, abnormality). The following definition reflects this:

Definition 3- Normal Subspace (Noncrisp Characterization):

The characteristic function of the normal (or abnormal) subspace is extended to take any value within the interval. In this case, the value represents the degree of normalcy: "1" indicates normal, "0" indicates abnormal, and the intermediate values represent elements with some degree of abnormality.2 The noncrisp characterization allows a more flexible distinction between normalcy and abnormality. However, in a real system, it may be necessary to decide when to raise an alarm or not. In this case, the problem becomes again a binary decision problem. It is easy to go from the noncrisp characterization to the crisp one by establishing a threshold.

2. PC Approach:

In this approach, we used the positive samples to build a characterization of the space. In particular, we did not assume a model for the set. Instead, we used the positive sample set itself3 for a representation of the space. The degree of abnormality of an element is calculated as the distance from itself to the nearest neighbour in the set. We chose to define the characteristic function of the set, since its definition is more natural, and the derivation of the set characteristic function is straightforward.

In a dynamic environment, the parameter values that characterize normal system behavior may vary within a certain range over a period of time. The term represents the amount of allowable variability in the self space (the maximum distance that a point can be from the samples to be considered as normal). This PC can be implemented efficiently using spatial trees. In our implementation, a KD-tree was used to represents a set of dimensional points and it is a generalization of the standard one-dimensional binary search tree.

The nodes of a KD-tree are divided into two classes: 1) internal nodes partition the space with a cut plane defined by a value in one of the dimensions 2) and external nodes (leaves) define "buckets" (resulting in hyper rectangles), where the points are stored. This representation allows answering queries in an efficient way. The amortized cost of a nearest neighbor query.

3. PC Experiments:

We performed experiments with real intrusion data obtained from the Lincoln Laboratory of the Massachusetts Institute of Technology. These data represent both normal and abnormal information collected in a test network, where simulated attacks were performed. The purpose of these data is to test the performance of intrusion detection systems. The data sets contain complete weeks with normal data (not mixed with attacks). This provides enough samples to train our detection system.

The test data set is composed of network traffic data (tcpdump, inside and outside network traffic), audit data (bsm), and file systems data. For our initial set of experiments, we used only the outside tcpdump network data for a specific computer (e.g., hostname: marx) and then we applied the tool tcpstat to get traffic statistics. We used the first week's data for training (attack free) and the second week's data for testing, which include some attacks. Some of these were network attacks, the others were inside attacks. Only the network attacks were considered for our testing. These attacks are described in Table I and the attack timeline is shown in Fig. 1.

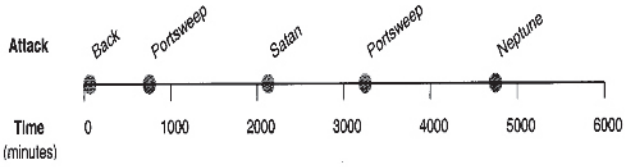

Fig.1. Attack timeline

Day	Attack Name	Attack Type	Start	Duration
1	Back	Dos	9.39:16	00:59
2	Portswep	Probe	8.44:17	26:56
3	Satan	Probe	12.02:13	02:29
4	Portsweep	Probe	10.50:11	17:29
5	Neptune	Dos	11.20:15	04:00

Table 1: Second week attacks description

Three parameters were selected to detect some specific type of attacks. These parameters were sampled each minute (using tcpstat) and normalized. Table 2 lists six time series and for training and testing, respectively. The set of normal descriptors is generated from a time series in an overlapping sliding window.

Name	Description	Week	Type
S1	No. of Buses/sec	1	Training
S2	No. of packets/sec	1	Training
S3	No. of ICMP packers/sec	1	Training
T1	No. of Buses/sec	2	Training
T2	No. of packets/sec	2	Training
T3	No. of ICMP packers/sec	2	Training

Table 2: Data sets and parameters used

In some cases, we used more than one time series to generate the feature vectors. In those cases, the descriptors were put side by side in order to produce the final feature vector. For instance, if we used the three time series S1, S2, and S3 with a window size 3, a set of nine-dimensional feature vectors was generated. In each experiment, the training set was used to build a KD-tree to represent the self set. Then, the distance (nearest neighbor distance) from each point in the testing set to the self set was measured to determine deviations.

For this set of experiments, the variables were considered independently, i.e., the feature vectors were built using only one variable (time series) each time. Fig. 2 shows an example of the training and testing data sets for the parameter number of packets per second.

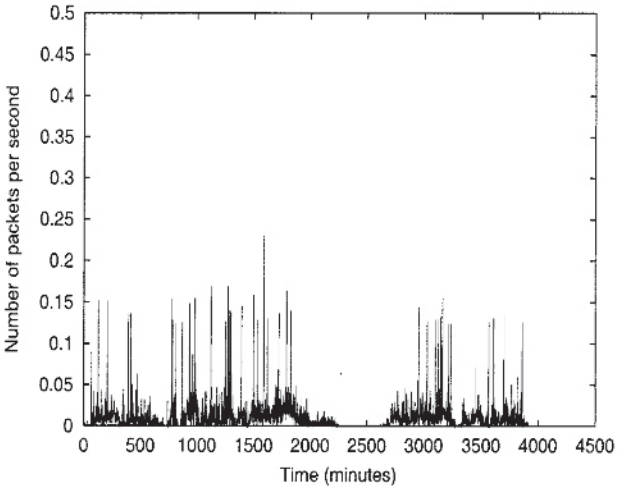

Fig. 2. Training and testing sets for the parameter no. of packets/Sec Training (self) set corresponding to the first week.

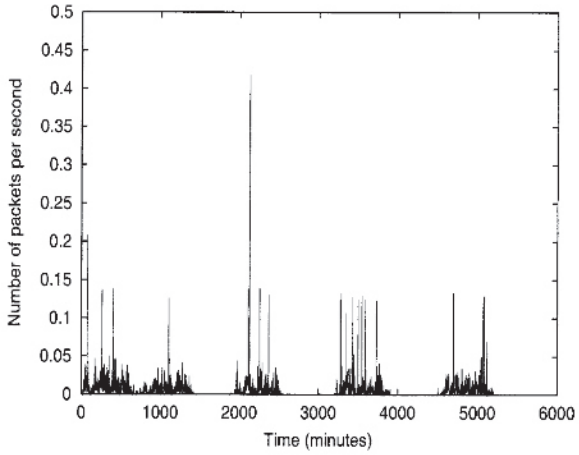

Fig. 3. Training and testing sets for the parameter no. of packets/Sec Testing set corresponding to the second week.

The representation of the characteristic function , i.e., the distance from the test set to the training set for the same parameter is shown in fig.4. In this case, the window size used to build the descriptors was 1. Fig.5 and 6 shows for using a window size of 3. In Fig. 5, the Euclidean distance is used and in Fig. 6 the distance is used.

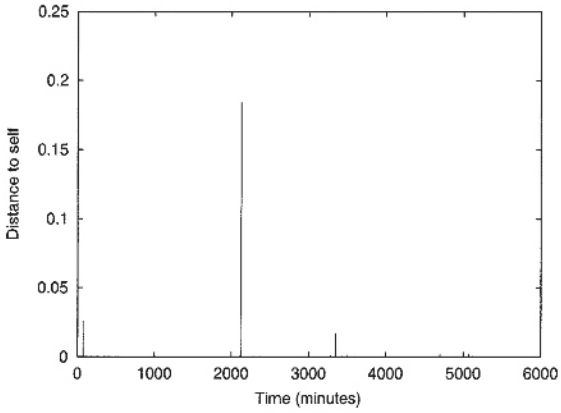

Fig. 4. Distance from the testing set (T2) to the self set (S2) (_ (~x)). For Window size 1 and Window size 3

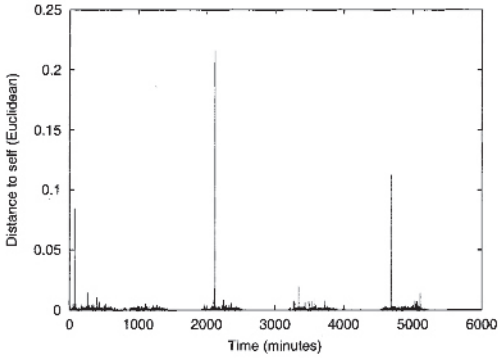

Fig. 5. Distance from the testing set (T2) to the self set (S2) (_ (~x)). For Euclidean distance

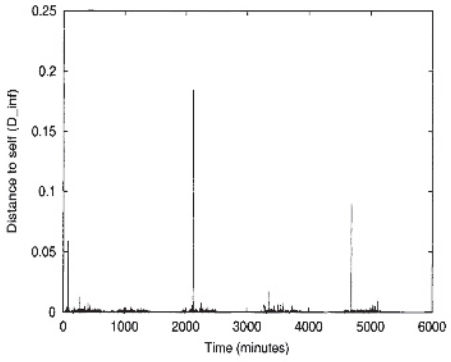

Fig. 6. Distance from the testing set (T2) to the self set (S2) (_ (~x)). For D distance

The plots of the nonself characteristic function show some peaks that correspond to significant deviations from the normal. It is easy to check that these peaks coincide with the network attacks present on the testing data (see Table 1). We conclude the following from the results.

1) Using only one parameter is not enough to detect the five attacks. Fig. 4 to 6 shows how the function detects deviations that correspond to attacks. However, none of the parameters is able to detect, independently, all five attacks.

2) A higher window size increases the sensitivity; this is reflected in the higher values of deviation.

3) A higher window size allows for the detection of temporal patterns. For the time series and , increasing the window size does not modify the number of detected anomalies, but, for the time series , when the window size is increased from 1 [see Fig. 4] to 3 [see Fig. 5 and 6], one additional deviation (correspondent to attack 5) is detected. Clearly, this deviation was not caused by a value of this parameter (no of bytes per second) out of range; otherwise, it would be detected by the window size 1.

4) The change of the distance metric from Euclidean [see Fig. 5] to [see Fig. 6] does not modify the number and type of the deviations detected. From the previous discussion, to detect the four attacks, it is necessary to take into account more than one parameter. In the following experiments, we used the three parameters to build the feature vector and test the ability of the system to detect the attacks. We performed two experiments varying the size of the sliding window:

4. Conclusion:

This level of deviation is compared with the distance range reported by the PC algorithm. Each row (and column) corresponds to a range or level of deviations. The ranges are specified on square brackets. The immune system has the property that foreign invaders (i.e., the nonself) are recognized easily with few false positives. This process proceeds in two stage s: in the first stage, the nonself is identified as a novelty (novelty detection) and the immune system then preserves a long-term memory for this pattern. By understanding the dynamics of the immune system, it is possible to implement a pattern recognition mechanism in the complement space where false positives and false negatives can be traded off as shown by ROC curves.

We investigated an immunocomputing technique to evolve novel pattern detectors in the complement pattern space to identify any changes in the normal behavior of monitored behavior patterns. This technique (NC) is used to characterize and identify different intrusive activities by monitoring network traffic and compared with another approach (PC). We used a realworld data set that has been used by other researchers for testing different approaches. The following are some preliminary observations.

5. References:

1. Mohammad Saniee Abadeh., Jafar Habibi., Zeynab Barzegar and Muna Sergi "A parallel genetic local search algorithm for intrusion detection in computer networks" Engineering Applications of Artificial Intelligence, Vol. 20, Issue 8, 2007, pp 1058-1069
2. D. Dasgupta,. F. Gonzalez, K. Yallapu, J. Gomez and R. Yarramsettii "CIDS: An agent-based intrusion detection system" Computers & Security, Vol. 24, Issue 5, Aug 2005, pp 387-398.
3. Shelly Xiaonan Wu and Wolfgang Banzhaf "The use of computational intelligence in intrusion detection systems: A review" Applied Soft Computing, Vol. 10, Issue 1, Jan 2010, pp 1-35.
4. Franciszek Seredynski and Pascal Bouvry "Anomaly detection in TCP/IP networks using immune systems paradigm" Computer Communications, Vol. 30, Issue 4, 2007, pp 740-749.
5. Gerry Dozier., Douglas Brown., Haiyu Hou and John Hurley "Vulnerability analysis of immunity-based intrusion detection systems using genetic and evolutionary hackers" Applied Soft Computing, Vol. 7, Issue 2, March 2007, pp 547-553.
6. Simon T. Powers and Jun He "A hybrid artificial immune system and Self Organising Map for network intrusion detection" Information Sciences, Vol. 178, Issue 15, Aug 2008, pp 3024-3042
7. Dipankar Dasgupta "Immuno-inspired autonomic system for cyber defense" Information Security Technical Report, Vol. 12, Issue 4, 2007, pp 235-241.
8. Azzedine Boukerche., Kathia Regina Lemos Jucá., João Bosco Sobral and Mirela Sechi Moretti Annoni Notare "An artificial immune based intrusion detection model for computer and telecommunication systems" Parallel Computing, Vol. 30, Issues 5-6, May-June 2004, pp 629-646.
9. Gerry Dozier., Douglas Brown., Haiyu Hou and John Hurley "Vulnerability analysis of immunity-based intrusion detection systems using genetic and evolutionary hackers", Applied Soft Computing, Vol. 7, Issue 2, March 2007, pp 547-553.