Intrusion Detection System Based on a Cooperative Topology Preserving Method

Emilio Corchado, Álvaro Herrero, Bruno Baruque, José Manuel Saiz

Department of Civil Engineering, University of Burgos, Spain.
E-mail: escorchardo@ubu.es

Abstract

This work describes ongoing multidisciplinary research which aims to analyse and to apply connectionist architectures to the interesting field of computer security. In this paper, we present a novel approach for Intrusion Detection Systems (IDS) based on an unsupervised connectionist model used as a method for classifying data. It is used in this special case, as a method to analyse the traffic which travels along the analysed network, detecting anomalous traffic patterns related to SNMP (Simple Network Management Protocol). Once the data has been collected and pre-processed, we use a novel connectionist topology preserving model to analyse the traffic data. It is an extension of the negative feedback network characterised by the use of lateral connections on the output layer. These lateral connections have been derived from the Rectified Gaussian distribution.

1 Introduction

The identification of intrusions is a difficult problem due to the dynamic nature of systems and networks, the creativity of attackers, the wide range of computer hardware and operating systems and so on.

This complexity increases if we talk about distributed network-based systems and insecure networks as Internet.

An attack and intrusion to a network would end up affecting any of the three computer security principles: availability, integrity and confidentiality, exploiting for example the Denial of Service, Modification and Destruction vulnerabilities [1]. Further, network intruders are constantly updating their attack technology.

For these reasons, intrusion detection systems have become a required element in addition to the computer security infrastructure of most organizations. Intrusion Detection (ID) is a field focused on the identification of attempted or ongoing attacks on a computer system or network. The accurate detection in real-time of computer and network system intrusions has always been a complicated and interesting problem for system administrators and information security researchers.

Intrusion Detection Systems (IDS) are software or hardware systems that speed up and automate the process of monitoring the events which take place in a computer system or network, analyzing them to identify security attacks.

There are two main models to analyze events for detecting attacks: Anomaly detection (identifies activities that diverge from known patterns for users) and Misuse detection (based on the comparison of a user's activities with the known behaviors of attackers attempting to penetrate a system) [2].

2 The Connectionist Analyzer Model

The Data Classification step used by this IDS model is based on the use of the neural architecture called Cooperative Maximum Likelihood Hebbian Learning [3, 4, 5]. It is based on the Negative Feedback Network [6]. Consider an N-dimensional input vector, $\mathbf{x}$, and a M-dimensional output vector, $\mathbf{y}$, with W_{ij} being the weight linking input j to output i and let η be the learning rate.

It can be expressed as:

$$y_i = \sum_{j=1}^{\mathbf{N}} W_{ij} x_j, \forall i \qquad (1)$$

The activation is fed back through the same weights and subtracted from the inputs.

$$e_j = x_j - \sum_{i=1}^{M} W_{ij} y_i, \forall j, \qquad (2)$$

After that simple Hebbian learning is performed between input and outputs.

Lateral connections [3, 4, 5] have been derived from the Rectified Gaussian Distribution [7] and applied to the negative feedback network. The net result [3, 4, 5, 8] will be shown to be a network which can find the independent factors of a data set but do so in a way which captures some type of global ordering in the data set.

We use the standard Maximum-Likelihood Network [3, 5, 9, 10] but now with a lateral connection (which acts after the feed forward but before the feedback). Thus we have: a feed forward step (Eq. 1) follows by:

Lateral Activation Passing: $$y_i(t+1) = [y_i(t) + \tau(b - Ay)]^+ \quad (3)$$

Feedback: $$e_j = x_j - \sum_{i=1}^{M} W_{ij} y_i \quad (4)$$

Weight change: $$\Delta W_{ij} = \eta . y_i . sign(e_j) | e_j |^{p-1} \quad (5)$$

3 IDS Model

The aim of this work is the design of a layered system capable of detecting anomalous situations for a computer network. The information analysed by our system is obtained from the packets which travel along the network. So, it is a Network-Based IDS [2]. The necessary data for the traffic analysis is contained on the captured packets headers. This data can be obtained using a network analyser.

When we talk about anomaly detection model we refer to IDS which detect intrusions by looking for abnormal network traffic. Anomaly detection is based on the assumption that misuse or intrusive behaviour deviates from normal system use [11, 12, 13]. In many cases, as in the case of the attacker who breaks into a legitimate user's account, this is a right assumption. The attacker may behave differently than the regular user, so if the IDS has established what the user normally does during a session, it can determine that the user is not behaving in a usual way and detect the attack.

So in summary, we have developed a system for detecting anomalous traffic patterns, this includes proper attacks and dangerous situations without being an attack. Examples of these ones are management actions performed by the network administrator, so in those cases, the administrator will know that is a real attack or just a false alarm in the case that he has performed it.

3.1 Structure of the Model

The structure of this novel layered IDS model is showed in Fig.1 and it is described as follows:

- **First step**. - Network Traffic Capture: one of the network interfaces is set up as "promiscuous" mode, in such a manner that it is capable of capture all the packets which are travelling along the network.
- **Second step**.- Data Pre-processing: the captured data is pre-processed and used as an input data to the following stage. We only select traffic based on UDP (User Datagram Protocol) as it is explained later. This means that in terms of TCP/IP (Transmission Control Protocol/Internet Protocol) protocol stack, the model analyses only the packets which use UDP at transport layer and IP protocol at network layer.
- **Third step**.- Data Classification: once the data has been captured and pre-processed, the connectionist model presented in section 2 is used to analyse the data and identify the anomalous pattern.
- **Fourth step**.– Result Display: the last step is related to the visualization stage. Finally the output of the network is presented to the administrator or person in charge of the network security. Up to the actual research state, this visualization tool displays data projections highlighting anomalous situations clearly enough to alert the network administrator as we show in Fig.2, taking into account aspects as the traffic density or "anomalous" traffic directions.

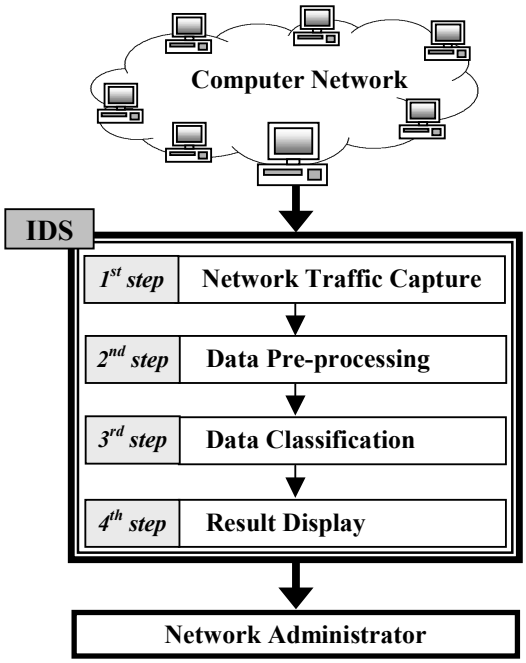

Fig. 1. Model Structure of the Layered IDS.

4 Real Data Set

The data pre-processing step is performed taking into account the following:

There are several protocols that can be considered dangerous for the network security: SNMP, ICMP, TFTP

and so on. Among those, we have actually focused our effort in the study of SNMP because an attack based on this protocol may severely compromise the systems security. CISCO [14] found the top five most vulnerable services in order of importance, and SNMP was one of them.

The study of SNMP protocol is the reason why the system selects packets based on UDP during the data pre-processing layer.

This research will continue trying to extend the model to cover several different situations, including other SNMP anomalous situations and protocols, until to cover all of them.

Data selection: we used only 7 variables extracted from the packet headers among all the information captured for each one:

- Timestamp: the time difference in relation to the first captured packet.
- Source Port: the port of the source host from where the packet is sent.
- Destination Port: the port of the destination host to where the packet is sent.
- Size: total packet size (in Bytes).
- Protocol: in this case we have used values between 1 and 35 to identify the packet protocol.
- Source IP: numeric value which codifies the source host IP address.
- Destination IP: numeric value which codifies the destination host IP address.

In terms of IP address, we have fixed numeric values to addresses included in each range in which the network is divided, given special values to the multicast and broadcast addresses.

This specific data set contains a scanning of network computers for the SNMP (Simple Network Management Protocol) port using sniffing methods. The aim is to make a systematic sweep in a group of hosts to verify if SNMP protocol is active in one of the following ports: 161, 162 and 3750. The sweep has been done using these port numbers because:

- 161 and 162 are the default port numbers for SNMP, as RFC 1157 [15] says: protocol entity receives messages at UDP port 161, and messages which report traps should be received on UDP port 162.
- We have also included a random port (3750) in the sweep as a test random element.

Some features of the analysed traffic along the network are the following:

- The SNMP packets are generated and sent inside the own network, this is, it is an internal protocol and any host out of the network can not introduce any packets of this type in the network. This is mainly warranted by the external security implemented through the firewall.
- We have taken into account all the traffic to ensure the existence of both, anomalous and non-anomalous situations. These have similar behaviours so the differences are difficult to identify making it an interesting problem to investigate.

5 Results

Fig. 2 shows traffic based on several protocols such as BOOTP, NETBIOS, DNS, TIMED and SNMP.

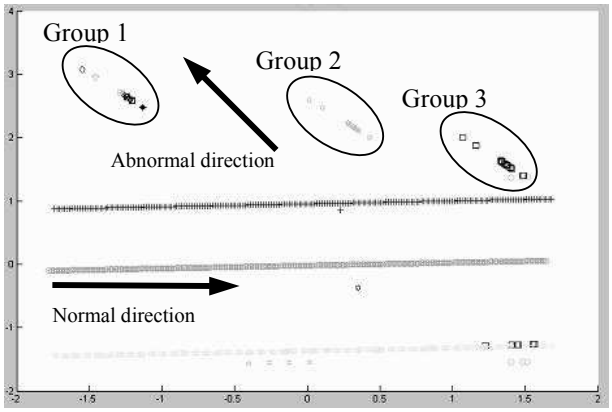

Fig.2. Data projections displayed by the model identifying anomalous situations.

Through a simple visual analysis of Fig.2 we can see that while most of the traffic evolves in the same direction, it is easy to identify three groups (Groups 1, 2 and 3.- Fig. 2) progressing in a different direction. We have study this matter (identifying every represented point) and we have concluded that these groups are related to the SNMP sweep mentioned above. Each group identified in Fig. 2 contains points that represent packets sent to each port included in the sweep (161, 162 and 3750) which is embedded in data set introduced to the model. All the packets belonging to SNMP protocol are contained in one of these three groups and there are no packets belonging to another protocol. In terms of performance results, our model has identified the three anomalous situations existing in the data set, as we known.

These graphical features allow the Network-Administrator to identify the sweep anomalous situation just by looking. The main feature that allows identifying the anomalous patterns is the growth direction. It can be seen that it is not parallel to the normal traffic direction.

6 Conclusions

We have developed a novel layered system for detecting anomalous traffic patterns including proper attacks and dangerous situations without being an attack, which can be considered an IDS. This work is actually focused on the study of SNMP because an attack based on this protocol may severely compromise the systems security.

We have applied different methods such as Principal Component Analysis [6, 16] or Maximum Likelihood Hebbian Learning for the classification step. Cooperative Maximum Likelihood Hebbian Learning provides more sparse projections than the others [5].

This is an ongoing research with the aim of showing the viability of the system developed. Later on it will be extended to cover a greater variety of anomalous situations as dictionary attacks or spoofing.

Future work will be based on the study of different distributions and learning rules to improve the whole architecture and to improve the system in such a way that it can be able to capture, process, classify and display the data in real time.

References

[1] Myerson, J.M. (2002) Identifying enterprise network vulnerabilities. International Journal of Network Management. 12(3).

[2] Planquart, J-P. (2002) Application of Neural Networks to Intrusion Detection. Information Security Reading Room - SANS (SysAdmin, Audit, Network, Security) Institute.

[3] Corchado, E., Fyfe, C. (2003) Orientation Selection Using Maximum Likelihood Hebbian Learning. International Journal of Knowledge-Based Intelligent Engineering Systems. 7(2).

[4] Corchado, E., Han, Y., Fyfe, C. (2003) Structuring global responses of local filters using lateral connections. J. Exp. Theor. Artif. Intell. 15(4): 473-487.

[5] Corchado, E., Fyfe, C. (2003) Connectionist Techniques for the Identification and Suppression of Interfering Underlying Factors. International Journal of Pattern Recognition and Artificial Intelligence. 17(8): 1447-1466.

[6] Fyfe, C. (1996) A Neural Network for PCA and Beyond. Neural Processing Letters. 6:33-41.

[7] Seung, H.S., Socci, N.D., Lee, D. (1998) The Rectified Gaussian Distribution. Advances in Neural Information Processing Systems, 10: 350.

[8] Corchado, E., Corchado, J.M., Sáiz, L., Lara, A. (2004) Constructing a Global and Integral Model of Business Management Using a CBR system. First International Conference on Cooperative Design, Visualization and Engineering (CDVE 04).

[9] Corchado, E., MacDonald, D., Fyfe, C. (2004) Maximum and Minimum Likelihood Hebbian Learning for Exploratory Projection Pursuit, Data mining and Knowledge Discovery. Kluwer Academic Publishing. 8(3): 203-225.

[10] Fyfe, C., Corchado, E. (2002) Maximum Likelihood Hebbian Rules. European Symposium on Artificial Neural Networks.

[11] Lunt, T., Tamaru, A., Gilham, F., Jaganathan, R., Neuman, P., Jalali, C. (1990) IDES: A Progress Report. Sixth Annual Computer Security Applications Conference.

[12] Denning, D. (1987) An Intrusion Detection Model. IEEE Transactions on Software Engineering. SE-13(2).

[13] Debar, H., Becker, M., Siboni, D. (1992) A Neural Network Component for an Intrusion Detection System. IEEE Symposium on Research in Computer Security and Privacy.

[14] Cisco Secure Consulting. (2000) Vulnerability Statistics Report.

[15] Case, J., Fedor, M.S., Schoffstall, M.L., Davin, C. (1990) Simple Network Management (SNMP). RFC-1157.

[16] Oja, E. (1989) Neural Networks, Principal Components and Subspaces. International Journal of Neural Systems. 1: 61-68.