Information-Theoretic Profiles for Intrusion Detection in LAN Traffic

Pablo Velarde-Alvarado[1], Alberto Martínez-Herrera[2], Adalberto Iriarte-Solís[1]

[1]Universidad Autonóma de Nayarit, Ciudad de la Cultura "Amado Nervo", Tepic, Nayarit, Mexico. Tel. +52 (311) 211 88 21, email: {pvelarde, adalberto.iriarte}@uan.edu.mx
[2]Instituto Tecnológico y de Estudios Superiores de Monterrey, Ave. Eugenio Garza Sada 2501 Sur, Monterrey, Nuevo Leon, Mexico. Tel. +52 (81) 8158 2269, email: a00798620@itesm.mx

ABSTRACT

In this paper, we show two methodologies for generating entropy-based behavior profiles of LAN traffic. Fast detection of intrusions caused by port scanning and worm attacks can be reached through the rate remnant elements at the packet-level, as well as the three-dimensional spaces of entropy at the flow-level. The profile was generated on LAN traffic of a campus by using empirical analysis.

Keywords: Intrusion detection, traffic profiling, entropy, and network worms.

1. INTRODUCTION

Network Intrusion Detection Systems [1], or NIDSs, have become an important component for detecting attacks against complex information systems. However, they offer only a limited defense. For instance, a signature-based NIDS (S-NIDS) monitors packets on the network and compares them against a database of signatures or attributes from known malicious threats. A weakness of this type of NIDS is that there will always be a lag between a new threat being discovered and the signature for detecting that threat being applied to the NIDS. During that lag time the NIDS would be unable to detect the new threat.

A second type of NIDS is the anomaly-based NIDS (A-NIDS), which monitors network traffic and compares it against an established baseline, [2]. The baseline helps to identify what is a normal behavior for that network. If a deviation from the established baseline reaches a specified threshold, an alarm is generated. Therefore, anomaly detection techniques have the potential to detect new and unforeseen types of attacks. Typical A-NIDSs employ algorithms that focus primarily on changes in the traffic volume at specific points on the network, and promptly alert the operator of a sudden increase. However, such systems can be evaded through sophisticated attacks that focus on compromising significant hosts, causing them a collapse of memory or CPU and maintaining a level of traffic within the normal threshold.

Currently, new generation of A-NIDS has emerged, which focus on gaining knowledge in the structure and composition of the traffic and not just its volume. Such systems are based on the fact that the malicious activities affect the natural randomness of the network, e.g., they change significantly the entropy of the network [4, 5]. The composition of traffic is related to its probability distribution, and can be characterized by its entropy; a malicious activity changes that composition and shape of the distribution. Thus, its entropy is changed. By means of entropy measures to a set of traffic features, we can establish the profiles of normal activity of the network and determine intrusions to the system.

This paper presents an analysis at the packet and the flow level on traces obtained through measurements conducted in a campus network under real attacks of the Blaster [6] and Sasser [7] worms, as well as a port scan attack to the proxy server of that network. The captured traces during a week of normal operation, helped to develop a profile of normal behavior that is useful to be compared to attack conditions.

The paper is organized as follows. In section 2, we present our profiling approach and the context of this paper. Section 3 describes the test environment; section 4 and 5 explain the methodology: the rate of remnant items and spaces of entropy and results. Section 6 gives concluding remarks.

2. BACKGROUND

As we discussed in the introduction, there are two types of NIDS, S-NIDS and A-NIDS. The last one is based on anomaly behavior detection. A-NIDS can outperform S-NIDS performance because of the first one is able to detect zero-day attacks. There are different approaches to implement an A-NIDS. One of them is based on entropy. Entropy helps to describe random processes. Network traffic can be modeled as a random process. Each random process is characterized by a probability function. If we are able to determine the probability function of the traffic network, we are able to apply entropy analysis.

Entropy analysis gives us several advantages. One of them is related to traffic that has been previously sampled. Entropy is not fully affected by the sampling process, so the traffic analysis can preserve its main features [8]. In addition, we can tune a window-time size to obtain accurate measures of the network traffic behavior. Literature suggest to use normally $t = 0.5$ s for short time periods or $t = 60$ s for long time periods and it allows to detect anomalies that can be hidden in the network traffic[9]. The window-time can be treated as static or dynamic. The first one means that each window-time is not overlapped while the last one is moved when a new packet arises. Dynamic window-time allows computing entropy on-the-fly and it help to obtain accurate measures [10]. These approaches can be mixed with two types of classification of traffic, packet-treatment or flow-treatment. The former refers to obtain each feature for each packet. Then, such set of features are used to compute the entropy. Flow-treatment is related to classify each feature according to a feature key. A flow-packet is a set of unidirectional packet sequences that have common values on each header packet [11]. When the flows have been constructed, each flow is classified according to a chosen feature key.

Even though packet-treatment seems the most accurate, it is not recommended for networks with high volume of traffic. Instead of packet-treatment, flow-treatment is suggested. The main advantage of flow-treatment is that the amount of data obtained is classified in such way that the relevant data are extracted, so the redundant data are discarded.

The success of the techniques based on entropy depends mainly on how the traffic profile is characterized. A reliable network traffic profile will help to establish reliable threshold parameters that will allow detecting anomaly behavior with accuracy. It only will depend on its network traffic behavior and not by using external references as can be seen at S-NIDS. It is clear that a reliable network traffic profile should be free of anomalies.

2. PROFILING APPROACH

We propose two methods for the creation of profiles based on entropy. The analysis applies primarily to the packet-level for the method of the rate of remnant elements and to the flow-level for the spaces of entropy. Initially, there is a set of captured traffic traces corresponding to five days in typical working hours in an academic LAN. The traces have been inspected to be considered free of anomalies, so they may serve as a baseline.

We use traffic features to build the profiles. A traffic feature is a field in a header of a packet (at the packet level) or a field in a five-tuple (at the flow level), respectively. Four fields will be used: source address (*srcIP*), destination address (*dstIP*), source port (*srcPrt*), and destination port (*dstPrt*).

After the feature extraction, an essential part in the builder profile block is the measurement of entropy. For a discrete set of symbols $\{a_1, a_2, a_3, \cdots, a_M\}$ with probabilities p_i, $i = 1, 2, \cdots, M$, the entropy of the discrete distribution of a random variable X associated, is a measure of randomness in the set of symbols and represented as

$$H(X) = -\sum_{j=1}^{M} p_j \log_2 p_j, \ 0 \le H(X) \le H_{MAX} = \log_2 M \quad (1)$$

The relative uncertainty (RU) provides a measure of variety or uniformity that is independent of the sample size. For a random variable X RU is defined as, [3],

$$RU = \frac{H(X)}{H_{MAX}}, \quad 0 \le RU(X) \le 1. \quad (2)$$

$RU(X) \approx 1$ means that observed values of X are closer to being uniformly distributed, thus less distinguishable from each other, whereas $RU(X) \approx 0$ indicates that the distribution is highly concentrated.

3. EXPERIMENTAL PLATFORM

The worm propagation and port scanning were carried out on academic LAN which is subdivided into four subnets (192.168.1.0, 192.168.2.0, 192.168.4.0, and 10.253.253.0). There are 100 hosts running Windows XP SP2 mainly. One router (192.168.1.1) connects the subnets with 10 Ethernet switches and 18 IEEE 802.11b/g wireless access points. The data rate of the core network is 100Mbps. A sector of the network is left vulnerable for worm propagation, with ten no Windows XP service pack installed at all (192.168.1.104 – 113). In the experiments Blaster and Sasser worms where released in the vulnerable sector. The scanning port attack was observed on the proxy server (192.168.4.253).

3.1 Data-set and tools

The benign traffic traces in typical work hours for a period of five days were labeled with a number from one to five. The anomalous traffic for port scanning attack was labeled as 6-P1. Blaster and Sasser worm attacks were labeled as 6-P2 and 6-P4, respectively. The data-set was collected by a network sniffer tool based on *libpcap* library used by *tcpdump*, [12]. All traces were cleaned to remove spurious data using *plab*, a platform for packet capture and analysis, [13]. Traces were split into segments using *tracesplit* which is a tool that belongs to *Libtrace*, [14]. The traffic-files in ASCII format suitable for MATLAB processing were created with *ipsumdump*, [15]. The flow generation was done with *flowanalyzer*, a tool based on *perl* and developed by us.

4. RATE OF REMNANT ELEMENTS

Let χ be a traffic trace divided into m non-overlapping slots, each one with duration $t \le t_d$. An i-slot is formed by W_i packets, where $i = 1, 2, \cdots, m$. For each i-slot, four features are extracted in such way that we associate them with a value of r, namely $r = 1$ for source IP address, $r = 2$ for destination IP address, $r = 3$ for source TCP port, and $r = 4$ for destination TCP port. Let $\boldsymbol{S}$ be a finite sequence of $r = 1$ values or IP source addresses in a slot-i. This sequence with elements in an alphabet set A, is a function from $\{1, 2, 3, \cdots, |A|\}$ to A for some $|A| \ge 0$. The sequence $\boldsymbol{S}$ is denoted by $(a_1, a_2, a_3, \cdots, a_{W_i})$, and the length of $\boldsymbol{S}$ is W_i. The elements of $\boldsymbol{S}$ belong to an alphabet A with cardinality $M = |A|$. From A, a sorted set $A^{(o)} = \{a_1^{(o)}, a_2^{(o)}, ..., a_M^{(o)}\}$ is obtained. $A^{(o)}$ has the M-source IP addresses sorted by frequency in decreasing order. With the associated frequencies of A, we estimate a probability mass function (pmf) for a slot i

$$\hat{p}_j\left(a_j^{(o)}\right) = f_j / W_i,\ j = 1, 2, ..., M \tag{3}$$

where f_j computes the number of times where each $a_1^{(o)}$ arises in $A^{(o)}$. f_j satisfies $f_1 \ge f_2 \ge f_3 \ge \cdots \ge f_M$. The set $A^{(o)}$ is transferred to an iterative process **P** to create l subsets of $A^{(o)}$, denoted as $A^{(o,k)}$, $1 \le k \le l$ .This family of l subsets is shown in Eq. 4-6 and holds $A^{(o,k)} \setminus A^{(o,k+1)} = \{a_k^{(o)}\}$

$$A^{(o)} = A^{(o,1)} = \left\{a_1^{(o)}, a_2^{(o)}, a_3^{(o)}, ..., a_M^{(o)}\right\} \tag{4}$$

$$A^{(o,2)} = \left\{a_2^{(o)}, a_3^{(o)}, a_4^{(o)}, \cdots, a_M^{(o)}\right\} \tag{5}$$

$$\vdots$$

$$A^{(o,l)} = \left\{a_l^{(o)}, a_{l+1}^{(o)}, a_{l+2}^{(o)}, ..., a_M^{(o)}\right\} \tag{6}$$

At the k-iteration, the relative uncertainty of $A^{(o,k)}$ reaches a threshold β, i.e., $RU(A^{(o,k)}) > \beta$, we say that the iterative process **P** reached its latest iteration, and hence, $k = l$. An estimator of relative uncertainty for the ordered set $A^{(o,k)}$ in the k-iteration is defined in terms of its estimated pmf as:

$$RU\left(A^{(o,k)}\right) = \frac{\sum_{j=k}^{M} \hat{p}_j(a_j^{(o)}) \log_2 \hat{p}_j(a_j^{(o)})}{\log_2(M-k)}. \tag{7}$$

Selecting a $\beta \approx 1$, the resultant subset $A^{(o,l)}$ is closer to being uniformly distributed. Then, for a given β, and a number l of iterations carried out, it is possible to calculate the remnant R_i^r for a subset $A^{(o,l)}$. Generalizing this for an i-slot and a r-traffic feature we have the rate of remnant elements:

$$R_i^r = \begin{cases} M & \text{when all } \hat{p}_j(a_j^{(o)}) = \frac{1}{M}, \\ M - l & \text{for } l \ge 1 \end{cases} \tag{8}$$

In other words, R_i^r is the cardinality of the subset $A^{(o,l)}$. We found that this feature under normal conditions presents regularities that allow creating behavioral traffic profiles. Table 1 summarizes the R_i^r behavior with $\beta = 0.95$ for our data-set.

Through of mean, variance, the intensity factor (σ^2/μ), and maximum value we can define a threshold for normal behavior of R_i^r. For instance, by averaging the means of R_i^r and its maximum values during benign traffic, we can define an average threshold of 28.5 with a maximum of 114.8 units. We denoted these thresholds for each r by $T(R_i^1) = (28.5; 114.8)$, $T(R_i^2) = (31.7; 115.6)$, $T(R_i^3) = (92.0; 342.6)$, and $T(R_i^4) = (132.3; 542.2)$.

Fig. 1 shows the patterns R_i^1 and R_i^2 for benign traffic in Trace 5 and their variations are inside of standard behavior mentioned above.

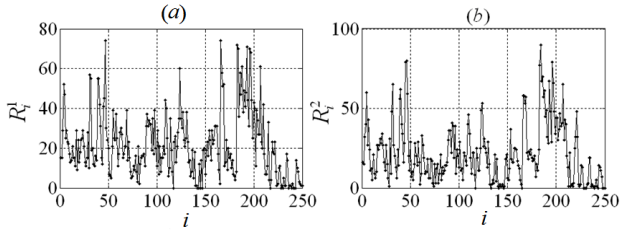

Fig. 1. Rate of remnants for (a) *srcIP* and (b) *dstIP* for standard traffic in Trace 5 in typical work hours. ($t_d = 60s$, and $\beta = 0.95$)

An anomaly related with a port scan attack directed to the proxy server was possible to detect it. It can be seen at the first slot where arose (i.e. i=2, 3) in trace 6-P1. The attack was carried out across a large number of TCP packets with source addresses supplanted. The growth of R_i^1 is possible to observe in Fig. 2(a) and is far away from $T(R_i^1) = (28.5; 114.8)$.

R_i^r patterns for worms attacks are presented in figures 2(b) and 2(c). There is a significant increase in the amount of remnants for $r = 3$ for Blaster and Sasser Worm propagation. It is important to highlight that the anomaly detection is done from the earliest slots where the intrusion arises.

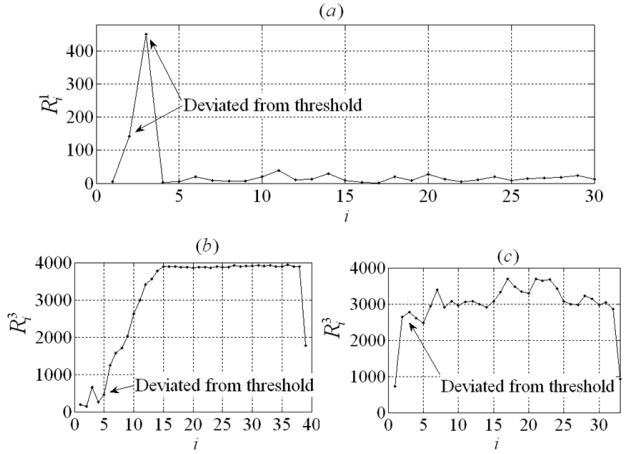

Fig. 2. Rate of remnant for: (a) *srcIP*, under port scan attack using spoofed IP addresses which is observable in slots i=2 and i=3 on Trace 6-P1 (b) *srcPrt* during Blaster

Worm on Trace 6-P2. (c) *srcPrt* during Sasser Worm on Trace 6-P4. All cases $t_d = 60s$ and $\beta = 0.95$

5. THREE-DIMENSIONAL SPACES OF ENTROPY

For each slot, the flows are stored on indexed text files. The traffic features applied for this technique are the flow's fields. They are labeled as $r = 1$ for source IP address, $r = 2$ for destination IP address, $r = 3$ for source TCP port, and $r = 4$ for destination TCP port.

Once that flows in an *i*-slot are generated, they should be clustered according to an *r*-flow feature. For a given *i*-slot, each cluster is formed by flows under the same r field value, but leaving freedom for the rest of r fields. For instance, taking $r = 1$ or key *srcIP* each cluster is formed by all the flows that contains the same source IP address field. It is regardless of the value on the rest of the fields $r = 2, 3, 4$. These are denoted as free-dimensions. The complete number of clusters will depend on the cardinality of the alphabet $|A_i^{r=1}|$, where $A_i^{r=1}$ is the alphabet of source IP addresses shown in slot *i.*

Thus, each cluster has flows with the same source IP address, but the rest of features ($r = 2, 3, 4$) have freedom of variation. In this context, we can estimate the entropy for each $r = 2, 3, 4$ on each cluster.

For each *i*-slot , the entropy estimation on free dimensions generates a cloud of data points in a 3-D Euclidean space, where the axis are $(\hat{H}_{\text{srcPrt}}, \hat{H}_{\text{dstPrt}}, \hat{H}_{\text{dstIP}})$ for $r = 1$. The points, $(\hat{H}_{\text{srcPrt}}, \hat{H}_{\text{dstPrt}}, \hat{H}_{\text{dstIP}})_1$, $(\hat{H}_{\text{srcPrt}}, \hat{H}_{\text{dstPrt}}, \hat{H}_{\text{dstIP}})_2, \ldots$ $(\hat{H}_{\text{srcPrt}}, \hat{H}_{\text{dstPrt}}, \hat{H}_{\text{dstIP}})|A_i^{r=1}|$ are plotted in the 3D-space. When we apply this procedure to the rest of cluster keys and all slots, we get four spaces of entropy.

Fig. 3 show the spaces of entropy. First, in Fig. 3(a), we see the point pattern for Trace-1, which corresponds to normal traffic conditions being typical for Traces 1 - 5. Fig.3(b) and 3(c) show a marked difference regard to benign traffic, since the data points move away from positions typically observed.

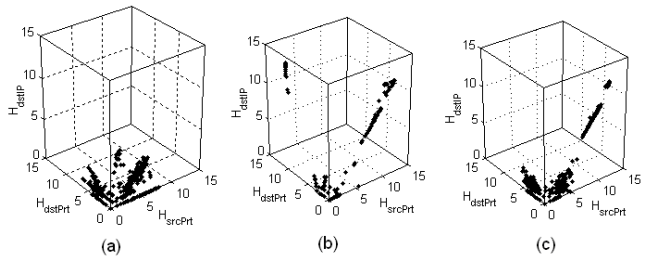

Fig. 3. Spaces of entropy for three srcIP cluster keys: (a) Traffic Trace -1 (benign traffic) in typical working hours. (b) Anomalous traffic, Trace 6-P2 (Blaster Worm) during 38 min period. (c) Anomalous traffic, Trace 6-P4 (Sasser Worm) during 38 min period

For a cluster key *r*, the characterization of entropy spaces under the vector $\mathbf{X}^r \in \mathbb{R}^3$ was performed by applying a technique of multivariable analysis. Such technique is the Principal Component Analysis. PCA provides a roadmap for how reduce a complex data-set to a lower dimension $\mathbf{Z}^r \in \mathbb{R}^d$, $d \leq 3$. It helps to reveal the hidden and simplified structure that often underlay on it. PCA is mathematically defined as an orthogonal linear transformation that converts the given data to a new coordinate system. This is done in such way that the greatest variance by any projection of the data lies on the first coordinate (called the first principal component, PCA 1), the second greatest variance on the second coordinate, and so on, [16].

To obtain accurate analysis estimation, tools like Kernel Density Estimation (KDE) may be useful. This analysis was applied on data-points at the slot level on the PCA 1 by using a Gaussian kernel of 200 points, with a bandwidth of $h = 1.06\sigma J^{1/5}$, given by the Silverman's criteria, where *J* is the number of observations and σ is the standard deviation of the set of observations. KDE shows that the traffic slots have Gaussian bimodality behavior in its pdf. Each mode was labeled as principal mode and far mode respectively, as shown in Fig.4.

Fig. 4a shows that the densities of PCA 1 in benign traffic

Table 1. Values of mean, variance, and intensity factor for the rate of remnants

Trace	Mean				Variance				Intensity Factor			
	srcIP	*dstIP*	*srcPrt*	*dstPrt*	*srcIP*	*dstIP*	*srcPrt*	*dstPrt*	*srcIP*	*dstIP*	*srcPrt*	*dstPrt*
1	29.1	33.1	103.5	136.5	328.2	377.5	4,502	8,818	11.29	11.42	43.5	64.61
2	29.5	35.1	88.8	138.6	566.5	688.1	3,830	12,918	19.23	19.62	43.16	93.24
3	31.1	33.0	96.2	141.7	497.9	595.2	4,598	11,972	15.99	18.06	47.78	84.51
4	31.1	35.9	103.2	141.9	732.8	727.9	6,209	18,868	23.59	20.26	60.16	133
5	21.5	21.5	69.0	102.88	277.8	351.1	3,208	9,191	12.94	16.33	46.5	89.3
6-P1	14.5	16.6	76.1	74.5	145.1	166.8	4,916	4,958	10.0	10.05	64.6	66.53
6-P2	27.9	19,618	3,107	1,005	3,464	6.2e07	1.7e06	4.3e04	12.42	3,209	549.7	43.24
6-P4	3,149	5,214	3,045	2,243	97,058	1.2e06	2.4e05	2.4e05	310.9	237.5	90.14	111.24

traces present a clear regularity in their shapes. This pattern of behavior changes drastically for anomalous traffic traces, see Fig. 4b and Fig 4c.

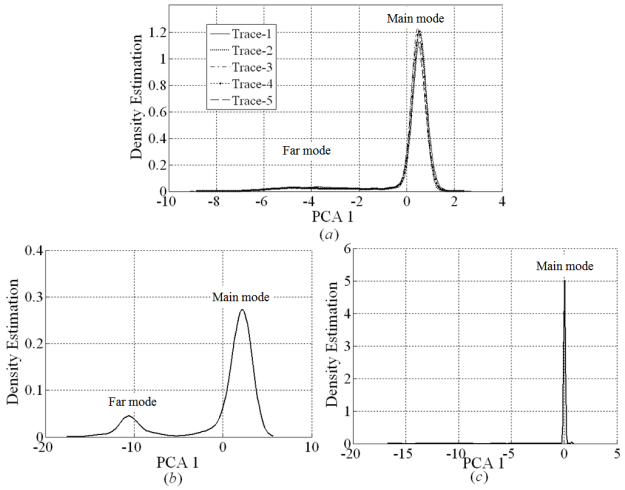

Fig. 4. Kernel density estimation for PCA1 for: (a) Traces 1-5 (Benign Traffic), (b) Trace 6-P2 (Blaster attack), and (c) Trace 6-P4 (Sasser attack)

This procedure was applied to *i*-slot level for a given *r*-, feature. The transformed data points are denoted by $\mathbf{Z}_i^r$. Densities of $\mathbf{Z}_i^r$ under normal conditions presents bimodality, the second mode (far mode) is situated on the left side of the main mode, at an average of minus five units. On the other hand, the empirically calculated average variance of $\mathbf{Z}_i^r$ for benign traffic is three units.

During analysis of the trace P1 was discovered that slots two and three had a variance of 0.32 and 0.74 units, respectively. These values are anomalous regards to the threshold of three units. Thus, an anomaly (Fig. 5a) was early detected in Trace 6-P1. The analysis of these traffic slots with Wireshark was able to identify a port scan attack.

In another case, Fig. 5b shows the first three anomalous slots, where far mode moves away from the typical value. The far mode displace to -9, -11 and -13 units, representing an anomaly with regard to the threshold of -5. This anomalous behavior in the far mode was caused by the spread of the Blaster worm.

Similarly, the Sasser worm attack caused anomalous values in the variance of the transformed data-points, These anomalous values were close to 0.5 units.

In all three cases studied, the attack leads to changes in traffic patterns that are detected from the first slots containing the malicious traffic.

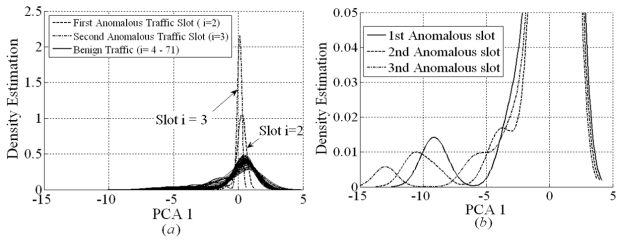

Fig. 5. Anomaly Detection on PCA1 (a) Anomaly caused by port scan detected in first slot in Trace 6-P1, (b) Anomaly caused by the Blaster worm that deviate the far mode out of the typical position

6. CONCLUSIONS AND FUTURE WORK

The generation of behavioral profiles based on entropy techniques offers an effective support for the Intrusion Detection Systems. The results of this study show that an A-NIDS employing generated profiles by the Rate of remnant elements or Three-Dimensional Spaces of Entropy methodologies can provide a rapid response for detecting deviations. It was performed against Blaster and Sasser worm-attacks as well as the port scanning, carried on in a campus network. The anomalies can be seen in the early slots where each attack arises.

As a future work, we will investigate the effect on variation of the slot duration t_d. Smaller values of slot duration represent faster response times, but also represent a smaller data set where to obtain representative traffic features. Finding the optimum value is an important objective design. Finally, we will study how to apply the presented methodologies with new types of network attacks.

7. ACKNOWLEDGEMENTS

The authors are grateful with PROMEP for financial support given for this research. A. F. Martinez-Herrera is very grateful with CONACyT and its Scholarship Funding Program support for pursuing his PhD studies at ITESM, Monterrey Campus.

8. REFERENCES

[1] D. Bolzoni and S. Etalle, **Approaches in Anomaly-based Network Intrusion Detection Systems**. Intrusion Detection Systems: Advances in Information Security. Springer Science+Business Media, LLC (2008)

[2] C. Kruegel, F. Valeur, and G. Vigna, **Intrusion Detection and Correlation**. Advances in Information Security. Springer (2005)

[3] K. Xu, Z. Zhang, and S. Bhattacharyya, **Profiling Internet Backbone Traffic: Behavior Models and Applications**. SIGCOMM 2005. (2005) 22-26

[4] A. Nucci, and S. Bannerman, **Controlled Chaos**. IEEE Spectrum. Vol.44. No.12. (2007) 42-48

[5] P. Velarde-Alvarado, C. Vargas-Rosales, D. Torres-Roman, and D. Munoz-Rodriguez, **Entropy Based Analysis of Worm Attacks in a Local Network**. Research in Computing Science. Vol. 34, (2008) 225-235

[6] D. Copley, R. Hassell, B. Jack, K. Lynn, R. Permeh, and D. Soeder, **ANALYSIS: Blaster Worm**. eEye Digital Security Research. http://research.eeye.com/html/advisories/published/AL20030811.html

[7] Y. Ukai, and D. Soeder, **ANALYSIS: Sasser**. eEye Digital Security Research. http://research.eeye.com/html/advisories/published/AD20040501.html

[8] Wagner A, Plattner B. **Entropy Based Worm and Anomaly Detection in Fast IP Networks**. Proc. of the 14th IEEE International Workshop on Enabling Tech (2005) 172 – 177.

[9] Velarde-Alvarado P, Vargas-Rosales C, Torres-Roman D. Martinez-Herrera A. **IP Traffic Anomaly Exposure, an Information Theoretic-based Approach**. Journal of Applied Research and Technology (2010) 159-176.

[10] Feinstein L, Schnackenberg, D, Balupari, R, Kindred, D. **Statistical approaches to DDoS attack detection and response**, DARPA Information Survivability Conference and Exposition (DISCEX 2003) (2003) 303-314.

[11] Malagón C. NetFlow y su aplicación en seguridad. *Ed. Red.es/RedIRIS*. (2009) 33-42

[12] V. Jacobson, C. Leres, and S. McCanne, **Tcpdump/libpcap**. http://www.tcpdump.org/

[13] A. Peppo, **plab. Tool for traffic traces**. http://www.grid.unina.it/software/Plab/

[14] Trac Project. Libtrace. http://www.wand.net.nz/trac/libtrace

[15] E. Kohler, **ipsumdump. Traffic tool**. http://www.cs.ucla.edu/~kohler/ipsumdump

[16] I.T. Jolliffe, **Principal Component Analysis**, Series: Springer Series in Statistics, 2nd ed., Springer, (2002), XXIX, 487 p. 28