Detecting Security Anomalies from Internet Traffic using the MA-RMSE Algorithms

Breno Pinto[1], Varin Khera[2], Chun Che Fung[3]
[1.] Computer Security Incident Response Team, BrasilTelecom, Brasília, DF, Brazil
[2.] Nokia Siemens Network, Australia
[3.] School of Information Technology, Murdoch University, Australia
Email: { [1]breno.silva@gmail.com, [2]varin.khera@nsn.com, [3]l.fung@murdoch.edu.au}

***Abstract* - Many detection techniques against worms, denial of service attacks and botnets on the Internet have been developed. It is difficult to detect these threats if the malicious traffic has insufficient intensity, which is usually the case. To make the problem worse, legitimate Internet services behaving like worm and complexity network environments undermines the efficiency of the detection techniques. This paper proposes an entropy-based Internet threats detection approach that determines and reports the traffic complexity parameters when changes in the traffic complexity content may indicate a malicious network event. Based on the experiment, the proposed method is efficient and produces less false positive and false negative alarms with a faster detection time.**

I. INTRODUCTION

Today, a reliable information and communication system is essential to the smooth operation of most of the organizations. However, the world still witnesses recurring events such as virus, worms and cyber attackers for numerous reasons, despite the significant research and activities in the discipline of Information Security. This illustrates the inherent weaknesses in the current information and communication technologies and an urgent need to develop a better and more proactive approach to achieve heightened security for present day and next generation networks. In this paper, a proactive anomaly detection method based on network behavior is proposed. The proposal is called Measure of Anomaly – Representative Measure of String Entropy (MA-RMSE). In this method, the detection of network anomalies is achieved by analyzing the current network bits distribution level. In general, it could be assumed that should a change of the traffic pattern is observed, this will indicate a possible attack. MA-RMSE uses a flexible and fast approach to estimate traffic distribution, by computing the entropy and statistical properties of the network traffic. Given the known history of normal traffic, it is possible to distinguish anomalies that change the Internet traffic bits distribution abruptly or slowly thereby indicating the occurrence of anomalies, and possible malicious activities.

Normally, entropy-based algorithms designed for Internet threats detection require a significant amount of malicious packets which cause rapid changes in the traffic pattern, before the detection algorithms are able to report the anomaly among the traffic [5]. This however is inefficient in detecting new forms of attacks, this paper therefore aims at presenting a new algorithm that provides a faster detection time and improved accuracy. This paper is organized as follows. Section II provides an introduction to the background of the entropy and detection algorithms. Section III discusses the implementation and related issues. Section IV presents some results based on some known attacks. Section V concludes the paper with discussion on further work.

II. ENTROPY AND DETECTION ALGORITHMS

The MA-RMSE technique is based on the measurement of the complexity of the Internet traffic. This is an application of the Information Entropy approach [1] and the resultant parameters will define the statistical properties of the traffic. These values in turn can be used to differentiate between normal and anomaly traffic patterns. In the following section, the general entropy theories of Shannon entropy, and the Kolmogorov complexity, which form the foundation of the proposed MA-RMSE algorithm, are introduced.

A. Shannon Entropy theory

Entropy is a measurement of the uncertainty associated with a random variable [1]. The term by itself in this context usually is referred to as *Shannon Entropy*. This value quantifies in the form of an expected value, the information contained in a message, usually in units such as bits.

Entropy *H(x)* can be described as:

$$H(x) = - \quad p(x) Log p(x) \qquad (1)$$

Where *Log* is the logarithm in base 2, which determines the degree of chaotic distribution of probability *p* and *x* is a string of bits. Traditional entropy traffic detection mechanism looks for the distribution of source addresses, destination addresses, ports, flow and correlates them to detect an anomaly [2,3,4]. This method works well in detecting distributed type of attack such as denial of service attacks, port scans, and large worm propagations pattern. However, the method is insufficient in detecting modern attacks which do not change the traffic patterns abruptly. Also, detection becomes difficult when the attack uses more than one protocol types and changes between the ports, and incurred other variations [5].

978-1-4244-3760-3/09/$25.00 © 2009 IEEE

B. *Kolmogorov Complexity theory*

Kolmogorov Complexity is about information and randomness. It deals with the amount of information from individual objects and it is measured by the size of its lower algorithmic description [6]. The measure of algorithm description is directly proportional to the random degree of strings of bits. For example, consider the following strings:

First String = "101010001100010101"
Algorithm description of First String = PRINT "101010001100010101"

Second String = "1111111111111111"
Algorithm description of Second String = PRINT "1 x 18"

In this example, it is shown that the First String is having a higher level of randomness and it therefore has a higher algorithm description than the Second String. In other words, First String is much more complex to be described. It is a concept of randomness and combines each binary string a numerical value that is considered as the "complexity" of the string. Kolmogorov Complexity *K(x)* can be described as:

$$K(x) = l(x)H(x) + Log_2(l(x)) \qquad (2)$$

where *l(x)* is the length of string, and x is a string of bits.

Kolmogorov Complexity can be defined simply as the size of the smallest program (or description algorithm) that computes the Turing Machine of a particular binary string [7].

On one hand, in the case that $K(x) \le |x| + c$, where $|x|$ is the length of string and *c*, as a constant asymptotically negligible, the string is described by an algorithm description smaller than its size. In an opposite case when $K(x) > |x| + c$, the string of bits has a high random degree and it cannot be described by a smaller algorithmic description.

The Internet traffic today is very complex in nature because there are large numbers of packets of different types and sizes carrying different services, therefore the entropy pattern of the Internet traffic is often high, except when an anomaly is observed. In other words, analyzing the Internet traffic for a specific service, the presence of high quantity of the same packets but with low complexity or entropy patterns could represent an anomaly. That is the scenario observed for example during worm propagation.

The Kolmogorov approach detects an anomaly using the formula described below:

$$K(p_1 p_2 p_3 ... p_n) < K(p_1) + K(p_2) + K(p_3) + ... + K(p_n) \qquad (3)$$

where p_x are the computed packets.

Figure 1 and 2 show the relationship between $K(p_1 p_2 p_3 ... p_n)$ and $K(p_1) + K(p_2) + K(p_3) + ... + K(p_n)$ during the normal behaviour patterns of the Internet traffic and distributed attack scenario. The convergence of these two functions could be observed during the anomalies behaviours. In other words, convergence will occur during the lower information entropy value of the analyzed traffic.

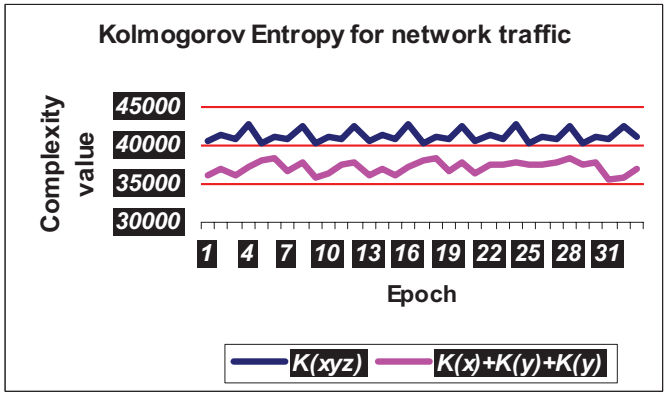

Figure 1: Kolmogorov Entropy during a normal Internet traffic.

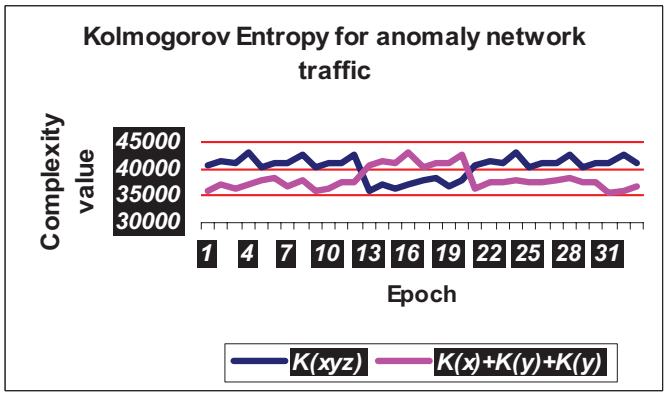

Figure 2: Kolmogorov Entropy during an anomaly Internet traffic.

Kolmogorov theory is an effective method to detect, identify and mitigate threats. However, it has one drawback as it requires a large number of malicious packets before it can detect an attack pattern. This has led to the development of MA-RMSE algorithm as described in the following section.

C. *MA-RMSE Algorithm*

The MA-RMSE proposed in this paper is a new algorithm in measuring the complexity characteristics of the Internet traffic. This algorithm draws its root from the Kolmogorov theory and it provides an efficient analysis of the bits distribution patterns which in turn is related to the quantity of the network traffic.

Two logarithmic functions are proposed and they are applied after a certain quantity of captured Internet traffic. In the given scenario, at least two hundreds packets by protocol and destination port are constantly computed and updated. Each packet stream provides information about the nature of the traffic and they will be related to each others, forming two curves with similar behavior. The basic idea is to use the proposed formula to measure the distribution of bit '1' patterns extracted from the packet.

The MA-RMSE algorithm formula are given below:

$$\alpha = \frac{Ln(\sum_{i=0}^{N} a_i + \sqrt{\frac{\sum_{i=0}^{N}(a_i + \bar{a})^2}{N-1}})}{Ln(2)} + \bar{a} \quad (4)$$

$$\beta = \frac{Ln((\left|3\mu_{1/2}(a) - 2\bar{a}\right|)N)}{Ln(2)} + (\left|3\mu_{1/2}(a) - 2\bar{a}\right|) \quad (5)$$

where N is the number of processed packets for a specific protocol port number, a is the quantity of bits *1* present in a string, $\bar{a}$ is the mean of bits *1* present in the processed packets, Ln is the natural logarithm function and μ is a variable that in this case is set to 1. In the future, this value will be adjusted so as to enable it to process the normal positive asymmetric traffic as described later.

These functions work by measuring the distribution of bits in a certain quantity of packets for each protocol port. Basically, when part of these processed packets decreases the entropy value of Internet traffic, the system will suspect the occurrence of anomaly or threat as $\alpha \geq \beta$. Similar to the probability theory and statistics, the mentioned relation between α and β is true for an anomaly situation because the distribution of bit 1 of computed packets is very close or it is a *normal distribution*.

With respect to Figure 3, it can be observed that the computed bits extracted from Internet traffic resulted in a negative asymmetric measure (Ad), $Ad < 0$, where $Ad = \alpha - \beta$, during almost all the time.

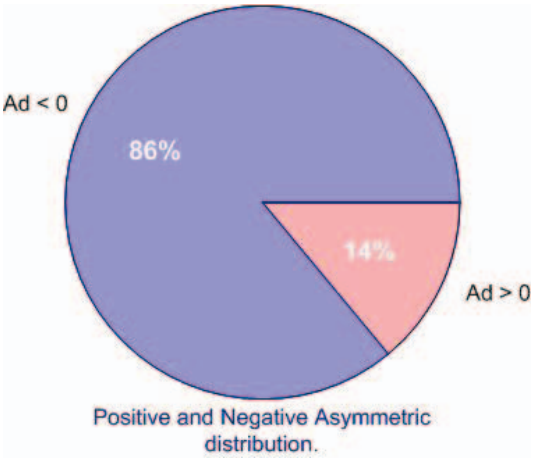

Figure 3: Negative and Positive Asymmetric distribution

This suggests that in the case of anomalies, the Internet traffic results in a symmetric measure $Ad = 0$ or changes from negative to positive asymmetric distribution $Ad > 0$.

To better understand this process the computed example for α and β is shown in Table 1 below:

# Packets	Number of bit 1 into each packet	α	β	Anomaly
11	100,100,100,100,100,100,100,150,250,300	146.98	35.50	Yes
11	47,69,82,100,146,168,179,201,250,289,310	178.27	180.13	No
11	150,150,150,150,150,150,150,150,150,150	160.68	160.68	Yes

Table 1: Example for MA-RMSE algorithm.

In addition to this, it is proposed that an *index number I,* is defined for each binary string when the algorithm detects an anomaly and a minimum quantity of the same *I* will help to reduce the false-positives, even in the case of positive asymmetric distribution of computed bits:

$I = 1_1 01_3 01_5 1_6 01_8 1_9 01_{11} 1_{12} 01_{14} 1_{15} 1_{16} 01_{18} 0 ... a_n$ if n is a number that represents the position of some bit *1* in the string, then $I = \sum_{i=0}^{N} n_i$. Each computed packet has an index *I*, and for an alert to be sent must exists at least five percent of same index I in a total of packets. This percentage sometimes requires adjustment for different type of services.

Figure 4 and Figure 5 show the relationship between α and β, during Internet traffic of a normal behavior of and distributed attack scenario respectively. The convergence of these two functions could be observed during the anomalies. In other words the convergence will occur during the lower information entropy value of the analyzed traffic.

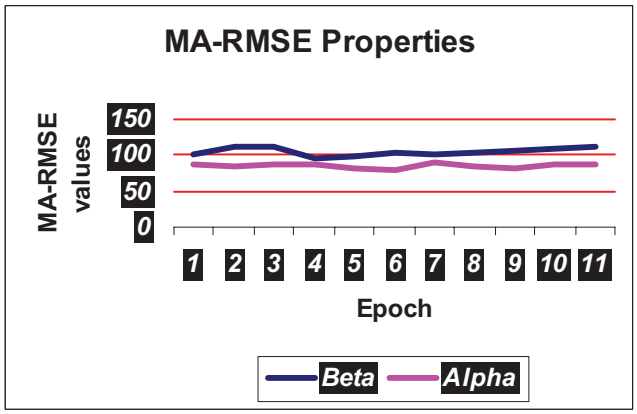

Figure 4: MA-RMSE properties during a normal Internet traffic.

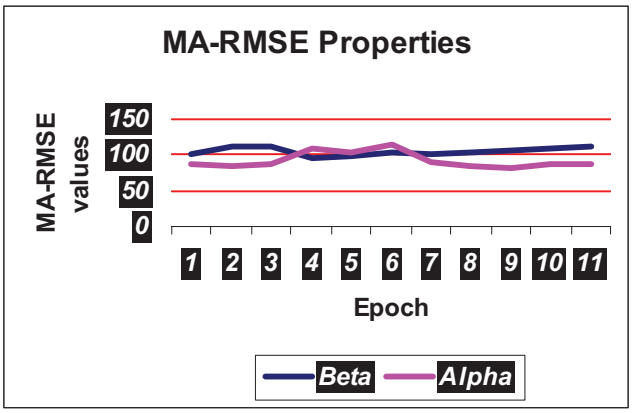

Figure 5: MA-RMSE properties during anomaly Internet traffic.

III. Implementation and Related Issues

The proposed detection algorithm has been implemented with C programming language using Libpcap [8] and PF_RING [9] has been used to collect data from the Internet. The program was installed on server connected to a two 1GB fibers Link with an estimated 400,000 packets per second which have been redirected to a capture port using a tap. This set up is outlined in Figure 6 below.

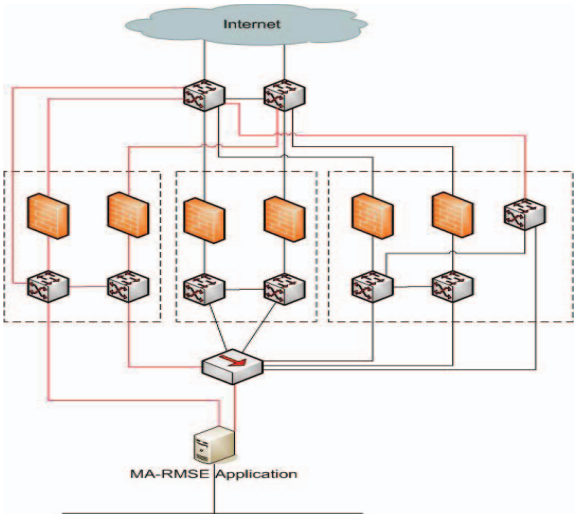

Figure 6: Network topology of the set up of the experiment.

The device has been positioned on the topology to process only the Internet traffic. To process the traffic, two *hash tables* [10] were created, both to allocate UDP and TCP packets.

Each structure can store two hundred packets for each port, then it processes those packets and in case of anomaly detection, an alert is sent to a syslog server containing Layer 2 to Layer 7 packet information.

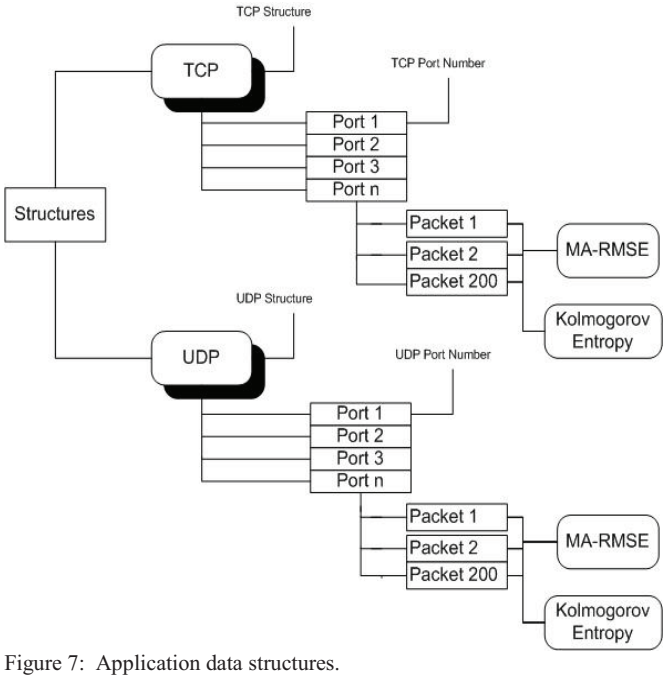

Figure 7: Application data structures.

The rule to decide between normal traffic and anomaly traffic is given below:

$$\begin{aligned} &\textit{If} && K(p_1p_2p_3) \le K(p_1)+K(p_2)+K(p_3) \\ &&& \textit{Anomaly} \\ &\textit{Else} \\ &\textit{If} && \alpha \ge \beta \textit{ and } \quad In \ge I\_Minimum \\ &&& \textit{Anomaly} \\ &\textit{Else} \\ &&& \textit{Normal Traffic} \end{aligned}$$

Initially, the proposed algorithm was used to inspect only 8 bytes of the incoming packets payload and it received numerous false-positives with *HTTP* and *BitTorrent* protocols, because with only 8 bytes for some protocols like HTTP all the packets seems to be the same.

The inspection of incoming packets is then increased to more than 40 bytes and the system did not receive any further false positive. Also, *l_Minimum* was defined as twenty as the minimum number of packets with the same index number. An example of the 8 and 40 bytes payload is given in Table 2 below.

Example of HTTP Google search payload	
8 bytes	GET /sea
40 bytes	GET/search?hl=ptBR&q=anomalies&btnG=Pesquisa+G

Table 2: Examples of 8 and 40 bytes payload.

After the bytes adjustment, it was noted that the test application took a few hours to detect the first threat without any false-positives.

IV. RESULTS

During the experiment based on traffic coming through a 1GB fibers link, some threats were detected. Most of these detected are publicly known, whereas a few others were unknown by the security community in general. Table 3 below gives the information about some of those detected threats from one of the experiments.

Threat/Attack Name	Type
Mainframe brute force attack	**Brute force**
X-R Botnet communication	**Bot**
DNS Dan Kaminsky Attack	**Massive Exploitation**
Massive URL access attack	**URL exhaustion**

Table 3: Threats detected in a two 1GB fiber link

All those threats and attacks detected had sufficient data into packet layer 7 to be computed by the algorithms and no signatures were used in the process.

V. CONCLUSION

A new MA-RMSE algorithm has been proposed in this paper and it has been tested with good initial success in detecting known and unknown threats from the Internet traffic. Some false-positives were initially observed during the experiment for *BitTorrent* and *HTTP* traffic but were later fixed through increasing the payload detection bytes. It is understood that there are numerous attack methods and botnet communication mechanism that uses less than 48 bytes of data therefore it was decided to allow different allocation of payload size for different TCP and UDP port. The optimal value of the number of bytes still has to be determined and it is currently under investigation.

REFERENCES

[1] Grünwald, Peter., Vitanyi, Paul. (2008) Shannon Information and Kolmogorov Complexity.

[2] Han, Chan-ky., Choi, Hyoung-Kee. (The School of Information and Communication Sungkyunkwan University, Suwon, South Korea - 2007) Entropy Based Worm and DDoS Attack Detection in Stub Networks

[3] Gil, T. and Poletto, M. (Vrije Universiteit, Amsterdam, The Netherlands - 2001) "MULTOPS: a data structure for bandwidth attack detection," USENIX 2001.rm and DDoS Attack Detection in Stub Networks

[4] Bazek, R., Kim, H., Rozovskii, B., and Tartakovsky, A. (2001) "A novel approach to detection of denial-of-service attacks via adaptive sequential and batch-sequential change-point methods," IEEE Systems, Man and Cybernetics Information Assurance Workshop, June 2001 NY – Paper WA 38.

[5] Wagner, Arno., Plattner, Bernhard, (Communication Systems Laboratory, Swiss Federal Institute of Technology Zurich - 2005) "Entropy Based Worm and Anomaly Detection in Fast IP Networks"

[6] Ming, Li., Vitanyi, Paul. (1997) An Introduction to Kolmogorov Complexity and its Applications. second edition" Springer, New York, USA.

[7] "Turing Machines".[Online], Available: http://plato.stanford.edu/entries/turing-machine/ (Accessed: May/2008)

[8] "Tcpdump/Libpcap".[Online], Available: http://www.tcpdump.org/ (Accessed: April/2008)

[9] "PF_RING". [Online], Available: http://www.ntop.org/PF_RING.html (Accessed: April/2008)

[10] Aaron M. Tenenbaum, Yedidyah Langsam, Moshe J. Augenstein. (1990), "Data Structures Using C, second edition", Pearson, New York, USA.

[11] "Demystifying Denial-Of-Service attacks, part one". [Online], Available: http://www.securityfocus.com/infocus/1853 (Accessed: April/2008)