I. J. Computer Network and Information Security, **2015, 9, 32-38**
Published Online August 2015 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijcnis.2015.09.04

Intrusion Detection Based on Normal Traffic Specifications

Zeinab Heidarian
Department of Computer Engineering, University of Isfahan, Isfahan, Iran
heidarian66@gmail.com

Naser Movahedinia
Department of Computer Engineering, University of Isfahan, Isfahan, Iran
naserm@eng.ui.ac.ir

Neda Moghim
Department of Information Technology Engineering, Faculty of Computer Engineering, University of Isfahan, Isfahan, Iran
n.moghim@eng.ui.ac.ir

Payam Mahdinia
Department of Electrical and Computer Engineering, Isfahan University of Technology, Isfahan, Iran
p.mahdiniaalvar@ec.iut.ac.ir

Abstract—As intrusion detection techniques based on malicious traffic signature are unable to detect unknown attacks, the methods derived from characterizing the behavior of the normal traffic are appropriate in case of detecting unseen intrusions. Based on such a technique, one class Support Vector Machine (SVM) is employed in this research to learn http regular traffic characteristics for anomaly detection. First, suitable features are extracted from the normal and abnormal http traffic; then the system is trained by the normal traffic samples. To detect anomaly, the actual traffic (including normal and abnormal packets) is compared to the deduced normal traffic. An anomaly alert is generated if any deviation from the regular traffic model is inferred. Examining the performance of the proposed algorithm using ISCX data set has delivered high accuracy of 89.25% and low false positive of 8.60% in detecting attacks on port 80. In this research, online step speed has reached to 77 times faster than CPU using GPU for feature extraction and OpenMp for parallel processing of packets.

Index Terms—Anomaly detection, one class support vector machine, false positive, GPU, OpenMp.

I. Introduction

Network intrusion detection systems are divided into signature and anomaly based detections. Signature based intrusion detection systems present higher accuracy in detecting known attacks. However, these methods are not able to detect unknown intrusions, so anomaly detection based on characterizing the normal traffic behaviour is utilised in such cases. The intrusion detection system is trained by the normal network behaviour and then any deviation from the normal behaviour is considered as an abnormality. However, achieving high percentage of anomaly detection is a challenging issue in this method.

To attain satisfactory performance, it is very important to select suitable features in learning the normal behaviour of the network traffic. Generally, the network anomaly detection techniques extract features from the packets' headers or exploit statistical specifications. Extracting the features from the header of the packets is deficient in detecting payload attacks. On the other hand, statistical techniques are very slow since they should model the payloads of packets. As payload anomaly detection has not received enough attention, extracting suitable features from the payloads of packets is still an open research topic.

This research shows that extracting features from both payloads and headers of the packets can lead to superior detection rate with higher speed compared to the statistical techniques.

Most of the existing anomaly detection techniques use DARPA99 data set to evaluate their results, but this data set has several drawbacks such as inappropriate selection of time to live (TTL) values for the packets [1]. Therefore, in this paper ISCX data set [2] is used to evaluate the performance of the proposed method. This data set provides attacks on payloads of packets without the drawbacks of DARPA99 data set.

It is very important for anomaly detection techniques to have high detection speed, because they should detect and alarm anomalies as they work online in an intrusion detection system. These techniques are very complex and need a huge amount of calculations, so their lack of speed is common. The detection speed has been increased in this paper by using GPU and OpenMp library to do parallel calculations. Extracting features from the

payloads of packets needs complex calculations, so GPU is used to speed up this procedure by employing multiple threads and doing parallel calculations. The detection phase of SVM algorithm is performed by OpenMp library, employing all CPU cores in parallel.

In the rest of this paper, Section two presents the related works. Section three describes the proposed method and its implementation. Section four shows that how detection time is decreased by using GPU and OpenMp. Section five is devoted to the results comparison with other methods. The performance evaluation results are reported in Section six. The proposed method is analysed in Section seven and the conclusions and suggestions for further improvements are presented in Section eight.

II. Related Works

There are several methods for payload anomaly detection which are offered to achieve high detection rate and low false positive results. Some of these are statistical and some others are based on feature extraction methods.

Payl method [3] models normal behaviour extracted from the payloads of the network packets. Byte frequency distribution and standard deviation of the normal packets entering a port is calculated during the learning phase. In the detection phase, Mahalanobis distance is calculated to determine the similarities between the actual data and the predetermined profile. If this distance is greater than a proper threshold value, anomaly detection system will warn. This method has been tested on DARPA99 data set and data set of Columbia University network. The results showed a nearly 100% detection rate with 0.1% false positive for the port 80's traffic. But as mentioned before DARPA99 data set is not a proper choice to evaluate the performance of an intrusion detection system.

In Anagram method [4], the correlation of a sequence of bytes in packets' payloads is calculated by modelling a sliding window of size n (a set of n consecutive bytes where n is greater than one) and therefore, skilful attacks could be detected. This method is designed to detect abnormal and suspicious packets. Anagram method uses bloom filter and binary based detection model. It doesn't calculate the frequency distribution of normal payload flows. During the learning phase, it generates its model by storing all separate windows of size n in a bloom filter without calculating the number of occurrences of these windows. After completing the learning phase, each packet will be scored by the windows of size n that have not been seen in the learning phase, as shown in (1).

$$Scre = \frac{N\ new}{T} \quad (1)$$

N is the number of new windows of size n that have not been previously seen and T is the total number of windows in the packet. In this method, windows are sliding windows with arbitrary length which are created on streams of bytes of every form of network traffic.

This method was tested on three selected web servers of Columbia University. The results showed detection rate of 100% for every type of worms and viruses seen on local area networks' traffic. Since this method is tested on specific and non-standard data set, the results are not applicable to any desired data set.

In Poseidon method [5], Payl architecture is changed to get better results. Poseidon is a method based on payload and with two layer architecture. First step is a Self-Organizing Map (SOM) and the second step is a variation of Payl system. Test results of this method on DARPA99 showed higher detection rate compared to Payl method. This method takes a packet as an input and classifies it regardless of the features such as packet length, port number and so on. The classification can be done by a neural network method, such as SOM. Fig. 1 shows the differences between Payl and Poseidon methods.

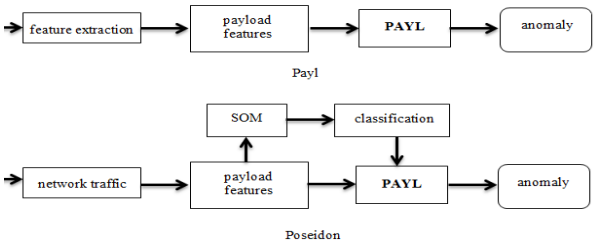

Fig.1. Differences between Payl and Poseidon method [5]

Acting as a kind of pre-processing, Payl is changed to Poseidon to use the advantages of unsupervised classification provided by SOM. Therefore, the results of Poseidon are better than Payl but using of DARPA99 data set is still a problem for evaluating the performance of this method.

Multiple classifier system for accurate Payload based Anomaly Detection (McPAD) [6] is another payload based system. Fig. 2 shows McPAD architecture.

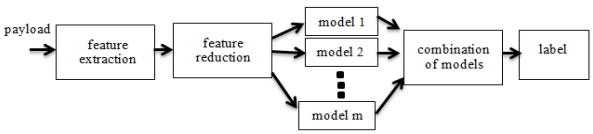

Fig.2. McPAD architecture [6]

As shown in Fig. 2, dimension reduction algorithm is run on the extracted features to achieve a combined model for normal traffic. Learning phase of this method is offline, so its complexity should be checked in the detection phase.

Results showed that neither Payl nor McPAD methods are able to detect unknown attacks with low error percentage.

Spectrogram, a machine learning method based on statistical anomaly detection sensor, is proposed in [7] to protect against the web layer attacks such as SQL, XSS and so on. Spectrogram collects packets continuously to reassemble their payload and recognize web layer allowed input. Spectrogram checks http requests and

models content and structure of the input strings and automatically learns the possible representation of the allowed inputs from the test data. Anomaly detection sensor checks the probability of the input string's permissibility.

In Spectrogram model a Markov chain is defined to model each normal input string based on probability of the characters of the string. Every chain is constituted from a number of states appointing for string characters. A transition between two states is the probability of a character existence in the condition of another character's presence in the string. Several Markov chains with different weights are considered to model each string and finally a linear combination of Markov chains is employed to determine the final probability. Then the string is compared with this model and any deviation is considered as an abnormal behaviour. Evaluation results showed that detection accuracy of this model especially for SQL injection and XSS attacks is 95%, but the speed of the method is very low due to the packet payload modelling.

Another statistical method is Geometrical Structure Anomaly Detection (GSAD) [8]. In this method, first network traffic packets are filtered based on some header features showing the occurrence frequency of one of the 256 ASCII characters in payloads of the packets. A model of http normal traffic is made by Mahalanobis distance map calculation and the weight metric is used to recognize the intrusion behaviour. This method has at most 0.15% false positive result. If the weight metric is greater than a predetermined threshold value, an alert will be generated. This method is very dependent on packet's destination address and its detection percentage will even be drastically reduced if this feature is deleted from the features list utilized to filter the packets

A cross domain collaborative method is recently proposed and tested on specific data set in [9]. In this method, the strings in normal traffic is divided to windows of size n, and then the hash functions of these windows are calculated and saved in a bloom filter. In fact, the normal traffic model is a bloom filter including the windows of size n of the monitored traffic. Payload based models are extracted by comparing the bloom filter as an array of bits. If at least 80% of windows of size n are matched with bloom filter, then the tested string will be considered matched to the bloom filter, otherwise an attack is alarmed. This system is able to detect the average of less than four new attacks with 0.03% false positive each day. Learning phase of this method is obtained from the 8 weeks data of three different servers in Columbia University. The disadvantage of this research is that the results are obtained from a specific data set.

Beside the methods previously explained, there are some other methods that use packet header features to detect anomaly. There are 41 header features in KDD99 data set. Birch method [10] uses these features and the first 37 features of KDD99 features set are extracted. Then a feature vector tree is dynamically made in the learning phase as the new samples of data enter the network. When the tree is made, a cumulative hierarchical clustering algorithm is directly used to show the nodes with their feature vectors. Then a centre is considered for each cluster to form a new set of clusters by calculating the distance of each data point to its nearest centre. When the feature vector tree is completely made in the learning phase, the nearest cluster is detected for each input of the tree in the testing phase. If its distance to the nearest cluster is less than the threshold, the input will be considered as normal behaviour and the tree will be updated accordingly. However, if the distance is greater than the threshold, the input will be considered as abnormal behaviour. The detection percentage of this method is 95%. In this method, only header features of the packets are extracted for anomaly detection, so this method couldn't be extended to detect the attacks on the payload of packets.

Intrusion detection system based on data mining technique [11] is another feature extraction based method. This method performs the clustering by means of Support Vector Machine algorithm (SVM) and is tested on NSL-KDD99 which is improved version of KDD99. This algorithm is one of the most successful clustering techniques in data mining field. Weka 3.7 software is used for data analyzing and Libsvm 1.5 for algorithm implementation. In this method, 41 features of KDD99 are extracted from each data set sample.

The results of this method including normal model learning time and detection percentage for different kernel functions are shown in Table1.

Table 1. The results of intrusion detection system using data mining technique

Kernel function type	Model learning time (sec)	Attack detection percent
Gaussian radial basis	77.01	98.57%
Polynomial	3859.57	98.43%
Sigmoid	615.75	73.09%

The results shown in Table 1 indicate that selection of Gaussian radial basis function has the detection percentage of 98.57% which can drastically decrease the learning time. But in this method, features are extracted only from the packets' headers.

In the following section, our proposed method will be explained. The aim of this method is to overcome the disadvantages of the previous methods and to achieve high detection accuracy and low false positive results.

III. The Proposed Method

In our method, 25 features are extracted from each packet payload and header as are described in section A. Then one class SVM algorithm [12] is run on these features in two steps: learning and detection.

The learning step is run offline while the detection step is run online. In section B, the offline step of the algorithm and in section C, the online step is described.

A. Extracted Features of the Packets

Since the packet payload anomaly detection is considered in the proposed method, most of the features are extracted from the payload of the packets and several features are extracted from KDD99. These 25 features are listed below:

1. The number of non-printable characters, with ASCII codes between 127 and 255. These characters are seldom used in normal packets' payload.
2. The number of small letters in the payload of packets.
3. The number of control characters in the payload of packets, with ASCII codes between 0 and 31 except 10 and 13. Normally the ratio of these characters to the characters between 31 and 127 which are mostly used in payloads is small.
4. The number of uncommon characters in the payload of packets such as $,{,},[,],/,|,\. These characters are used in buffer overflow attacks.
5. A Feature that shows the probability of the packet length normality according to the average length of normal packets. The greater this feature is, the more distance between URI length and normal URI length. This feature is calculated as in (2).

$$p(1) = \frac{\sigma^2}{(1-u)^2} \quad (2)$$

Where l is the URI length of the packet, σ is the URI length deviation and μ is the average URI length of the normal packets.

6. The number of packet defragmentation. Since attack packets are usually small and not broken into pieces.
7. The ratio of the number of big letters to the total number of characters in the payload of packets.
8. The ratio of the number of non-printable characters to the total number of characters in the payload of packets.
9. The ratio of the number of control characters to the total number of characters in the payload of packets.
10. The ratio of the number of numbers to the total number of characters in the payload of packets.
11. The ratio of the number of small letters to the total number of characters in the payload of packets.
12. The ratio of the number of uncommon characters to the total number of characters in the payload of packets.
13. The number of first type characters including characters with ASCII codes from 32 to 128 except 127.
14. The number of second type characters including characters with ASCII codes from 0 to 31and 127.
15. The ratio of first type characters to the total number of characters in the payload of packets.
16. The ratio of second type characters to the total number of characters in the payload of packets.
17. The number of particular words or phrases that appear in various attacks (especially XSS, SQL) and some phrases in attacks' packets' payloads such as the true equality of two same integer numbers (1=1), the phrase of "union select" (attacker retrieves table information by injecting the phrase of "union select <rest of injected query>"), the word of "user" (attacker tries the username many times to guess the valid username), the word of "exe" (attacker uses this word to execute illegal queries on database), the word of "convert" (an error message of illegal type conversion occurs as a result of this attack showing that the type of database is a SQLServer and the string value is of type conversion), the word of "shutdown" (attacker uses this word to shutdown database) and sequentially occurrence of one letter (attacker uses this method to overflow the buffer) [13].
18. The urgent flag being set.
19. The value of checksum in TCP packet. This flag is checked for identifying the damaged packets.
20. The size of packet header.
21. Total length of the IP packet.
22. The reset flag being set.
23. The finish flag being set.
24. The minimum value of the packet's TTL. When a stream is defragmented by an intruder, some defrags have small TTL values and will be lost in the network.
25. The standard deviation of character frequency in the payload of packets.

Features 1 to 17 and Feature 25 are extracted from the packets' payloads and Features 18 to 24 are extracted from the packets' headers.

B. Offline Step

As mentioned earlier the learning phase of the proposed algorithm is offline. In this section, 25 features are extracted from the packets of the flow in the network traffic, then one class SVM is trained with these features to make the normal traffic model.

C. Online Step

To detect anomaly in the test traffic, features of each input flow are extracted online and compared with the model made in the offline step, according to the one class SVM. If there is any deviation from the normal model, an alert will be generated. This online procedure is repeated for all of the traffic flows. The value of 0 or 1 is obtained for each traffic flow by using different kernel functions. 0 shows that the flow is normal and 1 shows that the flow is abnormal. If the selected kernel function is linear, the normal or abnormal value is calculated in (3).

$$u' \times v \quad (3)$$

Where u is the feature vector of the input traffic flow

and v is the feature vector of the normal model. If the selected kernel function is polynomial, the value is calculated in (4).

$$(\text{gamma} \times u' \times v + \text{coef}\,0)^{\text{degree}} \quad (4)$$

Where gamma, coef0 and degree are the kernel function parameters.

For the radial basis kernel function, (5) is used.

$$e^{(-\text{gamma} \times |u-v|2)} \quad (5)$$

If the kernel function is sigmoid, the value is calculated in (6).

tanh (gamma u v+ coef0)

$$\tanh\,(\text{gamma} \times u \times v + \text{coef}\,0) \quad (6)$$

Anomaly detection results will be different through the use of each kind of kernel functions described above. These results will be shown in the following section.

IV. Increasing Detection Speed Using GPU and OpenMp Library

In this research, the online step speed is increased by the means of parallel programming using GPU with multiple synchronized threads and OpenMp library. Extracting features needs complex calculations, so it's done by GPU. In the proposed method, extracted features from the packets can be divided into two groups; in the first group a certain string is looked for in the packets' payloads whereas in the second group other features are sought. The features of Group one will be extracted using FPAC algorithm [14] running on GPU to speed up the operation. But to extract the features of Group two, each thread of GPU solely processes only one packet [15].

SVM algorithm is computed by OpenMp library which uses all CPU cores for parallel calculations.

V. Comparison with previous methods

In this section the results of comparison between the performance of the proposed method and two of previous methods namely the spectrogram and the intrusion detection system using data mining technique are shown.

The proposed method and the intrusion detection system using data mining both use the same SVM algorithm for intrusion detection. The results of detection accuracy and learning time for different kernel functions are shown in Table 2.

As shown in Table 2, the intrusion detection system using data mining technique has higher detection accuracy and lower learning time for Gaussian Radial Basis function. But the learning time in the proposed method is lower when the polynomial and the sigmoid functions are used. The learning step speed is not very important, because this step is done offline and challenge of anomaly detection in intrusion detection systems is in online stage. In this research the online speed is improved using parallel programming.

Table 2. Comparison between intrusion detection system using data mining technique and proposed method

Methods	Kernel function types	Detection accuracy (percentage)	Model learning time (sec)
intrusion detection system using data mining technique	Gaussian Radial Basis	98.57%	77.01
	Polynomial	98.43%	3859.57
	Sigmoid	73.09%	615.75
Proposed method	Gaussian Radial Basis	79.13%	360.21
	Polynomial	77.34%	363.95
	Sigmoid	70.07%	364.93

The proposed method is also compared with spectrogram. Spectrogram method is reached to the desired results for the payload based anomaly detection. Therefore it is compared with proposed method from the perspective of detection accuracy, false positive and learning and detection time in Table 3.

Table 3. Comparison between spectrogram and proposed method

Methods	Detection accuracy (percentage)	False positive (percentage)	Detection time (sec)	Learning time (sec)
Spectrogram	88%	2%	3542.14	18549.73
Proposed method	79.13%	4%	174.37	177.25

Although the spectrogram method has higher detection accuracy and lower false positive, but its detection and learning speed are very low. So this method is not suitable for deployment in online IDS.

VI. Implementation and results

The file of Saturday 12/6/2010 with the size of 4.22 GB from ISCX data set which is normal traffic is used for the learning step and the file of Sunday 13/6/2010 with the size of 3.95 GB including attacks on packet payload is used for the detection step. First of all, 25 features are extracted from the normal file to construct the normal traffic model. Then each flow of the detection phase is checked online to detect the normality or abnormality. The results on the tested traffic with 3.95 GB in size are shown in Table 4, 5.

Speed of algorithm in the online phase for different kernel functions of one class SVM is shown in Table 4 for both CPU and GPU usage. Since the learning phase of the algorithm is offline, the learning time is not very important and the speed of algorithm should be tested in the detection step. As shown in Table 4, using GPU can drastically increase the detection speed. The accuracy and false positive results of the detection algorithm for

different kernel functions are shown in Table 5. According to Table 5, Gaussian radial basis function is suitable to be used because of its higher detection rate and lower false positive.

Table 4. Detection speed using CPU and GPU

Kernel function type	Detection speed using CPU (sec)	Detection speed using GPU (sec)
Linear	4757.34	57.15
Polynomial	4359.48	55.30
Gaussian radial basis function	4941.67	64.22
Sigmoid	2836.30	53.23

Table 5. Detection accuracy and false positive

Kernel function type	Detection accuracy (percentage)	False positive (percentage)
Linear	78.43%	9.70%
Polynomial	85.64%	11.52%
Gaussian radial basis function	89.25%	8.60%
Sigmoid	87.43%	9.70%

VII. Evaluation and Analysis

As said before, it is very important to achieve high detection percentage and low false positive in learning based anomaly detection. The proposed algorithm results showed about 89.25% detection accuracy, 8.60% false positive and very high detection speed of 64 seconds through the use of Gaussian radial basis function. Despite [3], [5], [6], [7], [10] and [11] tested on DARPA99 data set, the proposed algorithm is tested on ISCX data set. As this method achieves high detection accuracy and low false positive, it can be extended to be employed in intrusion detection systems. The proposed method has higher speed and accuracy in comparison with most of statistical methods which are used for payload anomaly detection such as Spectrogram. In our algorithm a feature based method is used to model the normal packets instead of statistical technique which has lower speed. Despite [4] and [9], which are tested on specific servers, the proposed method is customized to a specific hardware. Also in contrary to [8], [10] and [11] that only extract features from packet's header, a combination of features from header and payload of the packets are used in our method. So the accuracy of packet payload anomaly detection is improved.

VIII. Conclusion

In this paper, a method based on extracted features from packets' payloads and headers is proposed by the means of one class SVM algorithm. The evaluation of this method, tested on ISCX data set, showed the detection accuracy of 89.25% and 8.60% false positive with detection speed of 64 second for spotting anomaly caused by attacks on the packets' payloads of the test traffic. This method showed higher speed comparing to some statistical based anomaly detection techniques. The proposed method is employed to detect attacks on port 80, but could be extended for other protocols such as FTP, SMTP and Telnet as well.

References

[1] Brugger, S. Terry, and Ch. Jedadiah, "An Assessment of the DARPA IDS Evaluation Dataset Using Snort," Dept. Electrical Eng., Univ. California, 8th November 2005.

[2] The UNB ISCX 2012 dataset [Online]. Available: http://www.iscx.ca/dataset. [Accessed: 10 April 2013].

[3] J. Stolfo, K. Wang, and J. Salvatore, "Anomalous Payload-Based Network Intrusion Detection," in *proc. 2004 Symposium on Recent Advances in Intrusion Detection.*, pp. 203-221.

[4] K. Wang, J. Parekh, and J. Stolfo, "Anagram: A Content Anomaly Detector Resistant to Mimicry Attack," in *proc .2006 Recent Advances in Intrusion Detection*., pp. 020-06.

[5] D.Bolzoni, S.Etalle, and P.Hartel, "POSEIDON: a 2-tier Anomaly-based Network Intrusion Detection System," in *proc. 2006 IEEE International Workshop on Information Assurance*., pp. 10-156.

[6] R.Perdisci, D.Ariu, P.Fogla, G.Giacinto, and W. Lee, "McPAD: A Multiple Classifier System for Accurate Payload-based Anomaly Detection," in *proc. 2009 Computer Network, Special Issue on Traffic Classification and Its Applications to Modern Networks*, pp. 864-881.

[7] Y. Song, D. Keromytis, and J. Stolfo, "Spectrogram: A Mixture-of-Markov-Chains Model for Anomaly Detection in Web Traffic," in *proc. 2009 Annual Network and Distributed System Security Symposium*.

[8] A. Jamdagni, Z. Tan, P.Nanda, X .He, and R. Liu, "Intrusion detection using geometrical structure," *IEEE Frontier of Computer Science and Technology*, pp. 327-333, December 2009.

[9] N. Boggs, S.Hiremagalore, A.Stavrou, and J.Stolfo, "Cross-domain Collaborative Anomaly Detection: So Far Yet So Close," in *proc. 2011 International Conference on Recent Advances in Intrusion Detection*, pp. 142-160.

[10] K. Burbeck, "Adaptive Real-time Anomaly Detection for Safeguarding Critical Networks," Ph.D. dissertation, Dept. Computer and Information Science., Univ. Linköpings., February 2006.

[11] Y. B. Bhavsar and K. C. Waghmare, "Intrusion Detection System Using Data Mining Technique: Support Vector Machine," *International Journal of Emerging Technology and Advanced Engineering*, vol. 3, Issue 3, March 2013.

[12] H. Chih-Wei, Ch. Chih-Chung and L. Chih-Jen, "A Practical Guide to Support Vector Classification," Dept. Computer Science, Univ. National Taiwan, April 2010.

[13] W.G.J. Halfond, J.Viegas, and A. Orso, "A Classification of SQL Injection Attacks and Countermeasures," in *proc. 2006 IEEE International Symposium on Secure Software Engineering*.

[14] C. H. Lin, C. H. Liu, L. S. Chien, and S. C. Chang, "Accelerating Pattern Matching Using a Novel Parallel Algorithm on GPUs", *IEEE Transactions on Computers*, vol. 62, no. 10, PP. 1906-1916, October 2013.

[15] G. Vasiliadis, S. Antonatos, M. Polychronakis, E. P. Markatos, and S. Ioannidis, "Gnort: High performance

network intrusion detection using graphics processors," *Recent Advances in Intrusion Detection*, *Springer Berlin Heidelberg*, 2008.

Authors' Profiles

Zeinab Heidarian received the BS degree in Computer engineering from the University of Isfahan, Iran in 2010. She also received the MSC degree in Computer architecture from the University of Isfahan, Iran in 2013. Her research interests include anomaly detection as well as GPU usage for speed increasing.

Naser Movahedinia received his B.Sc. from Tehran University, Tehran, Iran in 1987, and his M.Sc. from Isfahan University of Technology, Isfahan, Iran in 1990 in Electrical and Communication Engineering. He got his PhD **.**degree from Carleton University, Ottawa, Canada in 1997, where he was a research associate at System and Computer Engineering Department, Carleton University for a short period after graduation. Currently he is an associate professor at the Computer Department, University of Isfahan. His research interests are wireless networks, signal processing in communications and Internet Technology **.**

Neda Moghim received the B.S. and M.S. degrees both from Isfahan University of Technology, Iran, Isfahan in 1999 and 2002 respectively and the Ph.D. from Amirkabir University of Technology, Iran, Tehran in 2009. She is the author of several technical papers in telecommunications journals and conferences. Currently she is an assistant professor at the Department of Information Technology Engineering, University of Isfahan, Iran. Her research interests are in the area of admission control and bandwidth management/ traffic engineering for QoS-enabled IP networks, next generation networks, and wireless networks.

Payam Mahdinia received the BS degree in Computer engineering from the University of Isfahan, Iran in 2010 and the MSC in Computer architecture from Isfahan University of Technology, Iran in 2013. He has three papers about using GPU power in accelerating intrusion detection systems in international conferences. His research interests include parallel processing, network security and computer architecture.

How to cite this paper: Zeinab Heidarian, Naser Movahedinia, Neda Moghim, Payam Mahdinia,"Intrusion Detection Based on Normal Traffic Specifications", IJCNIS, vol.7, no.9, pp.32-38, 2015.DOI: 10.5815/ijcnis.2015.09.04

