AN OVERVIEW OF FLOW-BASED AND PACKET-BASED INTRUSION DETECTION PERFORMANCE IN HIGH SPEED NETWORKS

HASHEM ALAIDAROS[1], MASSUDI MAHMUDDIN[1], ALI AL MAZARI[2]
[1]School of Computing, University Utara Malaysia, Malaysia
[2]Department of Information Technologies, Al-Faisal University, Prince Sultan College for Tourism and Business Jeddah, Saudi Arabia

Abstract

Network Intrusion Detection Systems (NIDSs) are widely-deployed security tools for detecting cyber-attacks and activities conducted by intruders for observing network traffics. With the increase in network speed and number and types of attacks, existing NIDSs, face challenges of capturing every packet to compare them to malicious signatures. These challenges will impact on the efficiency of NIDSs, mainly the performance and accuracy power.

This paper presents an overview of how the performance of the Payload-based and Flow-based NIDSs is affected by the threats and attacks within the high-speed networks environment. The impact of these new technologies on the NIDSs will be described in terms of the NIDSs performance and accuracy.

Throughout the analysis of the literature on this topic, we found that the Packet-based NIDSs process every packet (payload) received. While it produces low false alarms, it is very time consuming, therefore it is hard, or even impossible, to perform packet-based approach at the speed of multiple Gigabits per second (Gbps). Flow-based NIDSs have an overall lower amount of data to be process, therefore it is the logical choice for high speed networks but it suffers from producing high false alarms. Therefore, it can be recommended that, a hybrid or a mixture model of both NIDSs may ensure a higher ability to react on the wider scope of attacks within the high-speed networks environment.

Keywords: *Network Intrusion detection, packet-based, flow-based, high speed networks, efficiency*

1. INTRODUCTION

The number of Internet users is growing increasingly, along with new network services. Along with the wonderful benefits that the Internet gives, it also has its dark face. Since the Internet becomes bigger and bigger, network security attack threats have become more serious. Many security holes are exposed and misused by attacks. Recent reports on Internet security breaches indicate that the number and the damage cost are continuously rising [9]. Considering the damage cost caused by the attacks, it is important to detect attacks as soon as possible. For this purpose, Network Intrusion Detection Systems (NIDSs) have been developed.

There are two methods basis on the source of data to be analyzed in NIDSs: packet-based NIDSs and flow-based. Packet-based NIDSs has to analyze the whole payload content beside headers. In flow NIDSs, rather than looking at all packets going through a network link, it looks at aggregated information of related packets of network traffic in the form of flow, so the amount of data to be analyzed is reduced [9].

With the increase in network speed and number and types of attacks, existing NIDSs, face challenges of capturing every packet to compare them to malicious signatures. These challenges will impact on the efficiency of NIDSs, mainly the performance and accuracy power.

This paper presents an overview of how the performance and accuracy of the packet-based and flow-based NIDSs are affected by the threats and attacks within the high-speed networks environment.

This paper is organized as follows: Section 2 presents the concept and types of Intrusion Detection Systems (IDSs). This section also discusses about the IDSs efficiencies and challenges. Section 3 and 4 explain the concept of packet-based NIDSs and flow-based NIDSs, respectively. These sections also provide a discussion about the accuracy and performance in high-speed networks of packet-based and flow based. While section 5 shows the comparison between packet-based NIDSs and flow-based NIDSs, section 6 presents the conclusion of this work.

2. INTRUSION DETECTION SYSTEMS

Intrusion Detection can be defined as the process of monitoring and identifying the computer and network events, to determine the emergence of any abnormal incident, as consequence, this unusual event is considered to be an intrusion. It can be defined as “the process of identifying and responding to malicious activity targeted at computing and networking resources” [1]. It detects unwanted exploitation to computer system, both through the Internet and Intranet.

A reader may question how IDSs differ from firewalls. Firewall is defined as piece of hardware or software program which functions in a networked environment to prevent some communication forbidden by the pre-defined security policy. Firewall differ in the sense that they don’t usually have capability to search for anomalies or specific content patterns, such as spamming and worms, to the same degree as IDSs do. For these reasons, IDSs must be at first line of defense and work along with firewalls. Unlike firewalls, they are automated because they don’t depend on human’s decision. [31].

An Intrusion Prevention Systems (IPS) detects the attacks similar to IDS. An IPS is the next security layer that combines the protection of firewalls with the monitoring ability of IDSs to protect networks with analysis necessary to make the proper decision on the fly. IPSs, also named Intrusion Detection and Prevention Systems (IDPSs), are design to sit in-line with traffic flows and prevent attacks in real-time. Unlike IDSs (passive IDSs only raise an alarm in case of an intrusion), IPSs block traffics independently without human interaction. Therefore, the main disadvantages of IPSs are the serious consequences when blocking useful traffics (when false alarms rise) beside its bad performance in high speed network [29] . Since we focus on IDS, this paper doesn’t consider IPS.

2.1 IDS TYPES

2.1.1 HOST-BASED AND NETWORK-BASED

In general, we can divide IDSs into two basic classes based on their position in the network or audit source location: host-based IDSs (HIDSs) and network-based (NIDSs). HIDS monitors a single machine and audit data, such as resource usage and system logs, traced by the hosting operating system. On the other hand, NIDS, such as Snort, monitors a network and analysis the traffic which flows through the segment.

NIDSs have the following advantages: In contrast to HIDSs, the deployment of new host in network does not need more effort to monitor the network activity of that new host. Generally, it is easier to update one component of NIDSs than many components of HIDSs on hosts.

Since this paper focus on network flows and packets, we don’t consider host-based intrusion detection systems.

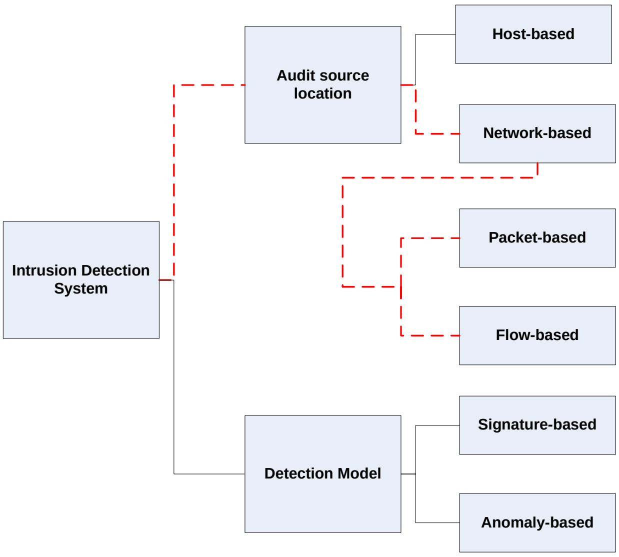

Figure 1: IDS Taxonomy, reproduced from [28] (dotted red line indicates the scope of our work).

2.1.2 SIGNATURE-BASED AND ANOMALY-BASED

IDSs also can be classified based on its detection model into two categories: signature-based and anomaly-based. The signature-based IDSs, also named “misused-based”, works similar to anti-virus software. It employs a signature (pattern that correspond to a know threat) database of know attacks, and if a successful match with current input, an alert is raised. A well-know example of this type is Snort [26] which is an open source IDS that monitors network by matching each packet it observes against a set of rules.

Anomaly-based or behavior-based IDS works by building a model of normal traffic data pattern during a training phase, then it compares new inputs to the model. A significant deviation (change) is marked as an anomaly (abnormal or intrusion). Although Signature-based IDSs cannot detect unknown attacks, either because the database is out of date or because no signature is available yet, it has low false alarm (high accuracy). On the other hand, anomaly-based is able to detect unknown attacks but it suffers from producing false alarms [10], [20]. Possible reasons for traffic anomalies are:

- Change in the network topology (e.g. routing changes or new connected hosts)
- Network usage (e.g., changed customer behavior, new applications)

2.1.3 PACKET-BASED AND FLOW-BASED

There are two methods basis on the source of data to be analyzed in NIDSs: packet-based and flow-based. Packet-based, also named "traditional NIDS", has to inspect the whole payload content beside headers. In flow-based NIDS, rather than looking at all packets going through a network link, it looks at aggregated information of related packets of network traffic in the form of flow, so the amount of data to be analyzed is reduced [29]. Thus, Flows provide information and patterns about network connection (can be represented in just a few bytes) to be analyzed rather than packet payload.

Packet-based mostly provides signature-based NIDSs valuable information to detect attacks while flow-based support anomaly-based NIDSs to have ability to detect anomalies [2] [19].

Details for these two methods are in Section 3 and 4. Figure 1 shows IDS types that has been discussed in this paper (dotted red line in the figure indicate the scope of our work). Readers interested in further taxonomy of IDSs, which are outside the scope of this paper, can refer to [18]

2.2 IDS EFFICIENCIES AND CHALLENGES

An efficient NIDSs has two features [10]:

- high accuracy (low false alarms)
- high performance (high speed of auditing)

Recent researches are still struggling to improve NIDSs to meet these two objectives. One of main requirement of NIDSs to achieve these objectives is scalability to high-speed networks. As network line speeds, as well as Internet traffic, continue to grow, the demand for faster security also increases. Most large corporations, universities, and government networks are moving toward speeds of up to 10Gbps. Researchers assess the current NIDSs processing potential to lie between 100Mbps and 200Mbps [14], [17]. Well-know NIDS like Snort [26] and Bro [23] show evidences of high resources consumption when deal with the overwhelming amount of data found in today's high-speed network [11].

The high speed of the network traffic means that the NIDSs will receive a large amount of data that should be monitored and the identified. The NIDSs' weak process ability is primarily because the analysis of network packets requires much computing. Thus, it is important to identify malicious packet in their early phases, which is feasible only in high speed routers [14]. Hence, it seems that the issue could be resolved by increasing the NIDS processor's speed. Unfortunately, the speed of networks increases faster than the speed of processors. It's not possible to keep up with the speed of network by just increase the CPU's speed of NIDSs [17]. In addition, storing the traffic for further analysis of the packet payload requires vast amount of storage area.

The other issue that confronts NIDSs is the growth and fast emergence of new attacks/viruses/worms on the Internet. In signature-base detection, as the number of attacks increase, the number of malicious (intrusion) signatures increase in the database in NIDSs. Usually, these databases contain hundreds or thousands of signatures. An NIDS has to add these new signatures into its signature list quickly without disturbing its main function of detecting intrusion. NIDSs then search for these signatures in network traffic to detect intrusions. To detect signatures, all network traffic must be compared with each and every signature to identify if a match exists or not.

This is very difficult particularly for today's high speed networks with line speed of 10 Gbps and beyond [14]. As a result, when packets are not analyzed on time, NIDSs start to drop packets as soon as they fail to compare all the coming packets with the signatures. These dropped packets may have aggressive data with attack signatures, which causes a high false negative (when no alert raise but intrusion attempt takes place) rates in NIDSs [24]

It is obvious from the issues mentioned in this section that NIDSs should be able to handle the growth in Internet bandwidth as well as the increase in line speed and the growing number of attacks. Thus, there is an urgent need for higher process ability of NIDSs.

3. PACKET-BASED NIDS

3.1 HOW PACKET-BASED WORKS

In packet-based NIDSs, all network packets passing a certain observation point such as a router are captured without any loss of information. The packet capture encompasses OSI (Open System Interconnection) layer 2 to 7. A packet has mainly two fields: the header which contains information about source, destination, and others, and the payload which contains the data.

In packet-based, also named "Deep Packet Inspection" (DPI), the combination of header and payload scan determines whether a packet is an intrusion or not. Generally, this approach is realized by making use of software such as tcpdump [30]. Incoming packets are scanned and every single rule of the database is checked against it as shown in figure 2. In addition, signature-based, that discussed in section 2.1.2, mostly apply packet-based process.

Packets capturing and analysis can take place at different locations such as routers, switches, and network monitors form which the resulting measurement data is transported to a remote analysis system.

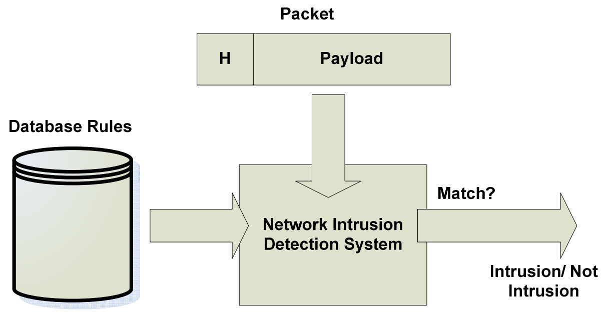

Figure 2: Packet-based NIDS, adopted from [29].

3.2 PACKET-BASED AND DETECTION ACCURACY

The main advantage of packet-based approach is that all common kinds of known attacks and intrusions practically can be detected if the data source deliver entire network packet for analysis. These ranges from attacks which are more connection-oriented to attacks occur only in network payload. Header information is mainly useful to recognize attacks aiming at vulnerabilities of the network stack implementation or scanning the operating system to identify active network services, On the other hand, payload information is most useful to identify active attacks against vulnerable application (since the connection that carries the attack is established in a normal way) [32]. However, the main disadvantage of packet-based is that it cannot detect unknown attack since it compare with predefined and known malicious signatures.

3.3 PACKET-BASED IN HIGH-SPEED NETWORKS

Since no information is lost in the monitored data, as we observed in section 3.1, it seems to be an advantage to this approach. But it is critical to note that more data does not automatically mean better results. The whole payload in every single incoming packet must be scanned. From the word “payload”, it implies that it “pays load” and overhead to the system. Every six months the network bandwidth is doubled [16], [15]. As a consequence, packet-based NIDSs must have high processing throughput so that they will be really fast and will not be the bottleneck for the network [16].

However, vast amount of data require vast amount of computational performance particularly complex algorithm in the domain of machine learning. In other words, systems that are capable of monitoring every packet on a high-speed network are very expensive and high resource consumption. Moreover, a drop of packets will occur if the NIDSs speed is not high enough to let the analysis process be done [27].

Packet-based scheme are very time consuming, therefore should not utilized in high-speed links, except when using high cost specialized hardware in specific network link. However, these hardware devices are quite expensive. In addition to the above problems, the amount of data needed for storing packet traces is usually huge and often has prohibitive costs. Therefore, the efficiency in accessing such database for analyses is also critical. Another issue facing packet-based in high speed network is that signature matching is impossible for most cases of encrypted payload, degrading the detection performance of NIDSs.

Since packet-based method is mostly applied in signature-based detection, the issues mentioned above apply in signature-based NIDSs as well. To handle with these problems, some approaches are evolved to reduce the amount of data to be identified that will be discussed in the following section.

3.4 RELATED WORKS

A significant amount of research has been done in order to design efficient packet-based NIDSs that perform its tasks in high speed network. Since usually not all incoming packets are related to intrusion, filtering and sampling approaches are the most common algorithm that applied into packet-based method.

Filters select packet depending on specific packet properties, such as values of packet header fields. Sampling algorithm, in general, aim at choosing a subset of packets which allows assuming information of the entirety of all packets, such as the frequency of a specific packet property. If packet sampling is combined with payload analysis, the goal is to remove packets where we don’t expect to find any interesting content, and to keep all packet of potential interest.

Another attempts to tackle the packet-based issues is [3], by dividing packet-based method into two stages to perform it in efficient way, called pre-filtering. The concept is that instead of matching the incoming packet against entire rule-set, it matches against sup-set of rules. More researches on improving packet-based technique in high speed networks can be found in [4] [13], [14], [17], [25].

4. FLOW-BASED NIDS

4.1 HOW FLOW-BASED WORKS

Flow-based technique is widely deployed as data source in applications like network monitoring, traffic analysis and security [21]. This method is characterized by flow data or network flow. One important fact about network flow is that flows don't provide any packet payload unlike packet-based approach. It rather relies on information and statistics of network flows, therefore flow-based NIDSs also called "network behavior analysis".

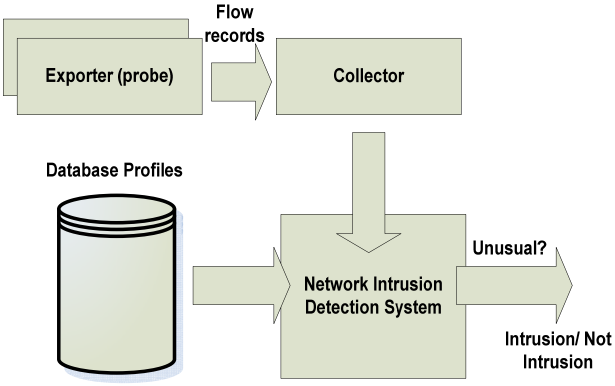

Figure 3: Flow-based components, reproduced from [5]

A flow can be defined as a unidirectional data stream between two computer systems where all transmitted packets of this stream share the following characteristics: IP source and destination address, source and destination port number and protocol value [5]. Nowadays special measurement systems are able to provide other characteristics in addition to the above, for instance:

- The number of packets and amount of bytes transferred in a flow
- The start and end time of a flow (in milli-second)
- The disjunction of all TCP flog occurring in the flow

These systems export these information in the form of Netflow [8] or IPFIX [7] records, called "flow record", to NIDSs to analyze them. These analysis systems can then be used to detect intrusion. Netflow and IPFIX are two protocols that specify the preparation and exportation format of flows. These software can be embedded in any Netflow-enabled switch or router.

A NetFlow setup consists of two components: an exporter and a collector. The flow exporter (can be a probe, a switch, or a router) extracts the headers from each incoming packet seen on the monitored interface. The exporter is responsible for creating flow records from observed traffic and sends them over the network to the collector. The collector stores these flow records, received from the exporter, and make suitable for NIDS for further analysis as shown in figure 3.

4.2 FLOW-BASED AND DETECTION ACCURACY

Since flows only provide information about behavior of connection and no packet payload, attacks which are detectable exclusively in packet-based cannot be detected. In other word, any attack that only injected in payload will not be identified in flow-based method. For example, popular web-based attacks which base upon the injection of malicious code into website can be invisible on flow-based. The attack patterns (malicious code) are only visible inside the packet payload. However, the attack might be detectable on flow-based if the attacker execute multiple attacks in parallel and causes many flows targeting the server [28].

Since flow-based are limited to information regarding network interactions, this information, however, is still possible to identify communication patterns between hosts. For many attacks this information is sufficient. In addition, many flow-based NIDSs have the ability to detect anomaly-based attacks. [2], [19] have presented an anomaly-based detection solution that relies on network flow-based data exported from NetFlow enabled-devices. There are intrusions which can easily be detected on flow-base. Such attacks are, Denial of Service (DoS), computer worms, network probing and flooding [28]. To understand how flows are used to detect attacks, you can refer to [28].

Moreover, unlike packet-based NIDSs, anomaly NIDSs using flow-based can detect unknown attacks. However, the main disadvantage of this technique is that it often suffers from increased false alarms.

4.3 FLOW-BASED IN HIGH SPEED NETWORK

Compared to packet-based method, flow-based deal with fraction of the total amount of data needs to be monitored and processed. For example, [28] calculated, in a high speed university network, that the ratio between packets exported by NetFlow (containing flow records) and the packet on the network is in average equal 0.1%. Furthermore, considering the network load measured in bytes, the overhead due to NetFlow is in average 0.2%. For this reason, performance issues in flow-based method are not a primary concern and therefore it is the logical choice for high-speed networks.

In addition, since flow-based data is very lightweight, the storage issues that appeared in packet-based approach are almost disappeared. Also unlike packet-based, encrypted payload does not influence the operability of flow-based NIDSs.

What must be kept in mind is that, as discussed in section 4.1, the flow records are generated in the exporter, therefore no performance overhead from computational resources occurs in NIDSs. It can be said that NetFlow largely offloads the task from the machine running the NIDS to the probe device [5]. For example, if routers are used for generating the flow records, its resources will be consumed on it, potentially having a negative effect on the routing itself and effecting overall network performance. To reduce this negative effect on the exporter, sampling process is used that supported by NetFlow. Thereby, only a subset of the packets is considered for flow generating, thus reducing the load on the router resources [6].

4.4 RELATED WORKS

For high speed networks, it is important to explore alternative to packet-based inspection for efficient NIDSs. One option that currently attracts the attention of researchers is flow-based intrusion detection. Since the absence of the payload contributes some advantages to flow-based such as scalability to high speed network, it also means that the flow-based NIDSs have difficulties identifying certain types of attack.

The issues mentioned above encourage researchers to enhance flow-based accuracy. Although researches in this area still relatively in its beginning, many of them achieved highly promising results in intrusion detection while focusing on flow-based data source especially to detect anomaly-based attacks.

In [28], it provides a comprehensive survey about current research in the domain of flow-based NIDSs. Mayung et al [22] suggests that by aggregating packets of the identical flow, one can identify the abnormal traffic pattern that appears during attacks. The two main examples of anomaly-based Denial of Service (DoS) detection in high speed networks, using flow-based only, are [19] and [33]. In [12], they developed a framework intended for detecting computer worms. Finally, [2] considered the effect of intelligent flow-based sampling techniques on anomaly detection.

5. COMPARISON BETWEEN FLOW-BASED AND PACKET-PASED IN NIDS

In table 1, we present a comparison of flow-based and packet-based of accuracy and performance aspects. To sum up, if we examine the shortcomings of both intrusion detection techniques, we discover a tradeoff between the flow-based NIDSs with only limited, aggregated data available for detecting intrusions and with higher speed of auditing, while in packet-based side we pay for additional data with higher resources consumption. We therefore expect a potential in mixing both approaches to detect at least the same quantity of attacks while consuming less resources.

Flow-based Intrusion detection	Packet-based intrusion detection
Flow records contain aggregated data up to transport layer (layer 4 in OSI)	Packets contain all complete payload and headers up to application layer (layer 7 in OSI). It therefore, considered more flexible in application of intrusion detection patterns.
Since the data availability is limited in NIDSs, defining accurate detection rules is not possible in all cases. This may result in a reduced alert confidence and higher number of false alarms.	Since the complete data is available, defining accurate detection can be on any part on traffic resulting in less false alarms and a higher alert confidence.
To process flow records, one must have an additional component of probe (exporter), see figure 3. On the NIDS side, the data must be first decoded. Therefore, the complexity is generally higher.	Packet-based NIDSs can be implemented easily without having any additional component to necessarily decode any protocol first.
The generation of flow records introduces a delay between the moment the first packet of a connection is established and the time when the record reaches the NIDSs. Depending on the configuration, record may be released after the connection has been closed o timed out.	The complete data is available to NIDSs immediately without delay.
Encrypted payload does not influence the operability of flow-based NIDSs	Signature matching is impossible for most cases of encrypted payload, degrading the detection performance of NIDSs.
Pre-filtering and aggregating flows are	Filtering and aggregating are

offloaded to the NetFlow probe device. The probe therefore must be measured adequately in order to reliably process amount of traffic flowing without lowering network performance.	maintained completely on the NIDSs machine itself. Pre-filtering is often implemented by using expensive hardware accelerate device such as FPGA.
Flow-based NIDSs have an overall lower amount of data to process, also in the analysis stage, because part of processing is outsourced to the probe device. Therefore, resource consumption is generally low.	Packet-based NIDSs, in most cases when no hardware pre-filter is used, must process every packet received, possibly generating a huge workload on the NIDSs. Therefore, resource consumption is generally high.
There are less privacy issues with flow-based method, as much of the potentially confidential content of connection never leave the transmission network.	Packet-based NIDSs receive the full payload data of every packet that may contain private data.

Table 1: Flow-based and Packet-based NIDS comparison, collected from [28], [5], [27].

6. CONCLUSION

This paper presents an overview of how the performance and accuracy of the Packet-based and Flow-based NIDSs are affected by the threats and attacks within the high-speed networks environment.

It was obvious from the literature that the current NIDSs should be able to handle the growth in Internet bandwidth as well as the increase in line speed and the growing number of attacks. Thus, there is an urgent need for higher process ability of NIDSs.

Packet-based NIDSs must process every packet (payload) received. While it produces low false alarms, it is very time consuming, therefore it is hard, or even impossible, to perform packet-based approach at the speed of multiple Gigabits per second (Gbps). Flow-based NIDSs have an overall lower amount of data to be processed, therefore it is the logical choice to work at high speed networks but it has less input information available to detect attacks and besides it suffers from producing high false alarms.

Basically, we could state a tradeoff between availability of limited data of flow-based techniques, which have negative effect on accuracy of NIDSs, and full data of packet-based which lead to a higher resources consumption. Therefore we believe that flow-based detection should not substitute the packet-based one in high speed environment, however, combination model to combine both approaches to power their advantages and overcome their drawbacks is suggested.

REFERENCE

[1] Amoroso E. Intrusion Detection: An Introduction to Internet Surveillance, Correlation, and Response, New Jersey, 1999.

[2] Androulidakis G., and Papavassiliou S. “Improving network anomaly detection via selective flow-based sampling,” Communications IET, pp. 399-409, 2008.

[3] Arelakis A., Efficient Pre-filtering Techniques for Packet Inspection, MSc thesis, Computer Engineering, Delft University of Technology, 2008.

[4] Brodie B., Taylor D., Cytron R., “A scalable architecture for high-throughput regular-expression pattern matching,” in ISCA, 2006.

[5] Cisco Systems, NetFlow Services Solutions Guide, http://www.cisco.com, 2011.

[6] Cisco, Random Sampled NetFlow. http://www.cisco.com, 2011.

[7] Claise, B, “Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of IP Traffic Flow Information,” RFC 5101, 2008.

[8] Claise, B., “Cisco Systems NetFlow Services Export Version 9,” RFC 3954 (Informational), 2004.

[9] Computer Economics, malware report: The economic impact of viruses, spyware, adware, bot-nets, and other malicious code, http://www.computereconomics.com, 2011.

[10] Debar H., Dacier M. and Wespi A. “Towards a taxonomy of intrusion-detection systems”, in Computer Networks, vol. 31, no. 8, pp. 805-822, 1999.

[11] Dreger H., Feldmann A., Paxson V., and Sommer R., “Operational experiences with high-volume network intrusion detection,” in Proceedings SIGSAC: 11th ACM Conference on Computer and Communications Security (CSS’04), pp. 2–11, 2004.

[12] Dubendorfer T., Wagner A., and Plattner B., “A Framework for Real-Time Worm Attack Detection and Backbone Monitoring,” In Proceedings of the First IEEE International Workshop on Critical Infrastructure Protection, pp. 3–12, 2005.

[13] Ficara D., Antichi G., Di P., Giordano S., Procissi G., and, Vitucci, F., "Sampling Techniques to Accelerate Pattern Matching in Network Intrusion Detection Systems,"

Communications IEEE International Conference, vol., no., pp.1-5, 2010.

[14] Gao M., Zhang K. and Lu, J. “Efficient packet matching for gigabit network intrusion detection using TCAMs,” in Proceedings of 20th International Conference on Advanced Information Networking and Applications (AINA’06), pp. 249–254, 2006.

[15] Ioannis S., Design and algorithms for packet and content inspection, Ph.D thesis, Computer Enginnering lab, Delft University of Technology, 2007.

[16] Ioannis S., Dimopoulos V., Pnevmatikatos D., and Vassiliadis S., “Packet pre-filtering for network intrusion detection,” in 2nd ACM/IEEE Symposium on Architectures for Networking and Communications Systems (ANCS), pp. 183–192, 2006.

[17] Lai H., Cai S., Huang H., Xie J., and Li H., “A parallel intrusion detection system for high-speed networks,” in Proceedings of the Second International Conference Applied Cryptography and Network Security (ACNS’04), pp. 439–451, 2004.

[18] Lazarevic A., Kumar V., and Srivastava J, “Intrusion detection: A survey,” Managing Cyber Threats, pp. 19–78, 2005.

[19] Le M., Tanaka Y., “Anomaly identification based on flow analysis,” TENCON IEEE conference, pp. 1-4, 2006.

[20] Morin B., and M’e L., “Intrusion detection and virology: an analysis of differences, similarities and complementariness,” Journal in computer virology, vol. 3, pp. 39-49, 2007.

[21] Muraleedharan N., “Analysis of TCP Flow data for Traffic Anomaly and Scan Detection,” 16th IEEE International Conference on Networks, 2008.

[22] Myung S., Hun K., Seong H., Seung C., and James H., “A Flow-based Method for Abnormal Network Traffic Detection,” IEEE/IFIP Network Operations and Management Symposium, pp. 599-612, 2004.

[23] Paxson V. “Bro: a system for detecting network intruders in real-time,” Computer Networks, vol. 31, no. 23–24, pp. 2435–2463, 1999.

[24] Salour M. and Su X. “Dynamic Two-Layer Signature-Based IDS with Unequal Databases”, Fourth International Conference on Information Technology: New Generations (ITNG 2007), Las Vegas, Nevada, pp. 77-82, 2007.

[25] Smith R., Estan C., Jha S., and Kong S., “Deflating the big bang: fast and scalable deep packet inspection with extended finite automata,” SIGCOMM CCR, 2008.

[26] Snort, intrusion detection system, http://www.snort.org, 2011.

[27] Sommer R., Viable Network Intrusion Detection in High-Performance Environments, PhD thesis, Informatics faculty, TU Munchen Univerisity, 2005.

[28] Sperotto A., Schaffrath G., Sadre R., Morariu C., Pras A., and Stiller B., “An Overview of IP Flow-Based Intrusion Detection,” Communications Surveys Tutorials, IEEE, pp. 1-14, 2010.

[29] Stephen N., and Judy N., Network Intrusion Detection, New Riders, 2003.

[30] Tcpdump, http://www.tcpdump.org, 2011.

[31] Tuck N., Sherwood T., Calder B., and Varghese G., “Deterministic memory efficient string matching algorithms for intrusion detection”. In Proceedings of the IEEE Infocom Conference, pp. 333–340, 2004.

[32] Wang K. and Stolfo S., “Anomalous Payload-Based Network Intrusion Detection,” Proceedings in 7th Symposium on Recent Advances in Intrusion Detection, volume 3224 of LNCS, pp. 203–222, 2004.

[33] Yan G., Zhichun L., Yan C., “A DoS Resilient Flow-level Intrusion Detection Approach for High-speed Network,”Proceedings of the 26th IEEE International Conference on Distributed Computing Systems (ICDCS’06), 2006.

Hashem Mohammed Alaidaros is a PhD student at School of Computing in University Utara Malaysia. He received his Bachelor in Computer Engineering from Jordan University of Science and Technology, Jordan in 2003. His Master degree was obtained in 2007 from University Putra Malaysia in Computer System Engineering. He is a lecturer since 2008 in the Information System Department, Al-Faisal University, Prince Sultan College, Jeddah, Saudi Arabia. His research interests include information security, security analysis of network traffic, and E-Commerce.

Massudi Mahmuddin obtained his PhD in 2008 in the areas of system engineering, Cardiff University, United Kingdom. He is currently a senior lecturer with Department of Computer Science, School of Computing, University Utara Malaysia. During last 10 years of his stay at the school, his teaching research and development interests have been in the areas of technical and social aspect of computing, computational intelligent and expert system.

Ali Al Mazari, earned his PhD in Science from the School of Information Technologies, University of Sydney in Australia, early 2007, received his Master of Computing (Major in IT) from the School of Computing, University of Western Sydney, Australia, 2001, received his Master of Leadership and Management in Education, University of Newcastle, Australia, 2010, and received his Bachelor of Science in Mathematics and Computer Science in 1994. Dr Ali is currently the Head of IT Department at Al-Faisal University, (Prince Sultan College in Jeddah Campus, Saudi Arabia, Jeddah). He is also the Director of the IT Centre at the PSCJ, KSA. His research interests include Machine Learning, Data Mining, Bioinformatics, Information Security, and Cybercrime Management.