Volume 4, Issue 12, December 2014 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

# Detecting Network Attacks Using Flow-Based Techniques

**Sudhir S. Kanade**
Prof. & Head E & T C Dept.
B.A.M. University

**Deepak V. Jadhav**
ME (Computer Sci. & Engg.),
B.A.M. University

***Abstract***— ***Intrusion detection is a crucial area of research nowadays. Earlier methods to detect the attacks are based on content verification of every packet flowing through the network. Packet inspection is very time consuming process and increases the overhead of intrusion detection system. Therefore, the researchers discover the alternative method named flow-based intrusion detection. Instead of verifying the packet contents, the flow of packet through the network is inspected.***

***The goal of this paper is to implement the flow based intrusion detection system as a supplementary service to the content-based intrusion detection. The paper provides the overview of the flow-based approach and shows how flow-based approach can be used to detect attacks.***

***Keywords— Network flows, intrusion detection, attacks, DoS, scan.***

## I. INTRODUCTION

An intrusion is the act of gaining unauthorized access to a system so as to cause loss or harm. Nowadays hackers are continuously attacking networked systems; in fact, it would be interesting to investigate if there are still Internet users who have not been victim of an attack yet. Considering the damage caused by the attacks (billions of U.S. dollars) [2], it is important to detect attacks as soon as possible, and take, if feasible, appropriate actions to stop them. It is a tedious task to perform due to the variations in form attacks exhibit. Generally, the intrusion detection systems are classified in two categories: Host based Intrusion Detection Systems (HIDS) and Network-based Intrusion Detection Systems (NIDS). Special systems have been developed to detect networks attacks based on NIDS. In an attempt to find known attacks or unusual behaviour, these systems traditionally inspect the contents (payload) of every packet [3], [4]. The problem of packet inspection, however, is that it is hard, or even impossible, to perform it at the speed of multiple Gigabits per second (Gbps) [5], [6]. For high-speed lines, it is therefore important to investigate alternatives to packet inspection. One option that currently attracts the attention of researchers is flow-based intrusion detection. With such approach, the communication patterns within the network are analyzed, instead of the contents of individual packets. Nowadays special measurement systems are able to provide, for every pair of IP addresses and port numbers, aggregated information, such as the time data exchange has started, the time it has stopped, the amount of transferred bytes and the number of sent packets. These systems export this information in the form of Net flow [7], [8] or IPFIX [9] records to systems that analyse them. These analysis systems can then be used to detect intrusions.

Flow-based detection can be seen as a complement of packet inspection, and should not be seen as a replacement. Both approaches can be combined into a two-stage detection process. At the first stage, flow-based approaches can be used to detect certain attacks. At the second stage, packet inspection can be used to additionally protect critical servers or selected systems, for which the first stage has discovered suspicious activities[1]. This paper proposes a system to detect the network attacks based on flow-based intrusion detection. Since we are using network flows as our main input, our paper does not focuses on host-based intrusion detection systems. The paper is organized as follows: Section II describes the motivations that have encouraged us to start this research. Section III explains the concept and ideas behind flows, as well as the network infrastructure needed for flow monitoring and analysis, such as intrusion detection. Section IV describes how flow-based approach can be useful to detect Denial of service and Scan attack, whereas Section V provides result of the proposed system. Finally, Section VII presents some conclusions and discusses the strengths and weaknesses of flow-based approaches.

## II. MOTIVATION

The Internet is a complex system in constant evolution. The common observations with respect to security like, the number of attacks continues to grow, the growth in Internet traffic as well as the increase in line speed, the spread of encrypted protocols poses a new challenge to payload-based systems.

Given these problems, flow based approaches seem to be a promising candidate for Intrusion Detection research. Flow-based Intrusion Detection Systems will analyse the information related to network interactions and detect attacks. Compared to traditional NIDS, flow-based NIDS have to handle considerable lower amount of data.

Sometimes it is argued that flows do not carry enough information, compared to pay load inspection, for being useful for intrusion detection. Though, the Flows are limited to information regarding network interactions, still it is possible to identify communication patterns between hosts, when communication takes place and which amounts of packets and bytes have been moved. For many attacks, this information is sufficient. In any case, it is important to underline that flow-based intrusion detection is not supposed to substitute the packet-based one, rather complements the approach by allowing early detection in environments in which payload-based inspection is not feasible [1]. As described by Schaffrathet al.[10], in an ideal world pay load-based solutions would always outperform flow-based one sin accuracy. In high-speed networks, however, the processing capabilities of the NIDS may be too limited to allow payload-based approaches.

## III. PROPOSED SYSTEM

"A flow is defined as a set of IP packets passing an observation point in the net work during a certain time interval. All packets belonging to a particular flow have a set of common properties."

Accounting flows is a two-step process: flow exporting, and flow collection. These tasks are performed by two components: flow exporter and flow collector. Figure 1 shows this exporting /collecting process.

The flow exporter, also known as observation point, is responsible for the metering process, i.e., creating flow records from observed traffic. The flow exporter extracts the packet header from each packet seen on the monitored interface. Each packet header is marked with the times tamp when the header was captured. After that the header is processed by a sampling- filtering module, where it can be sampled or filtered.

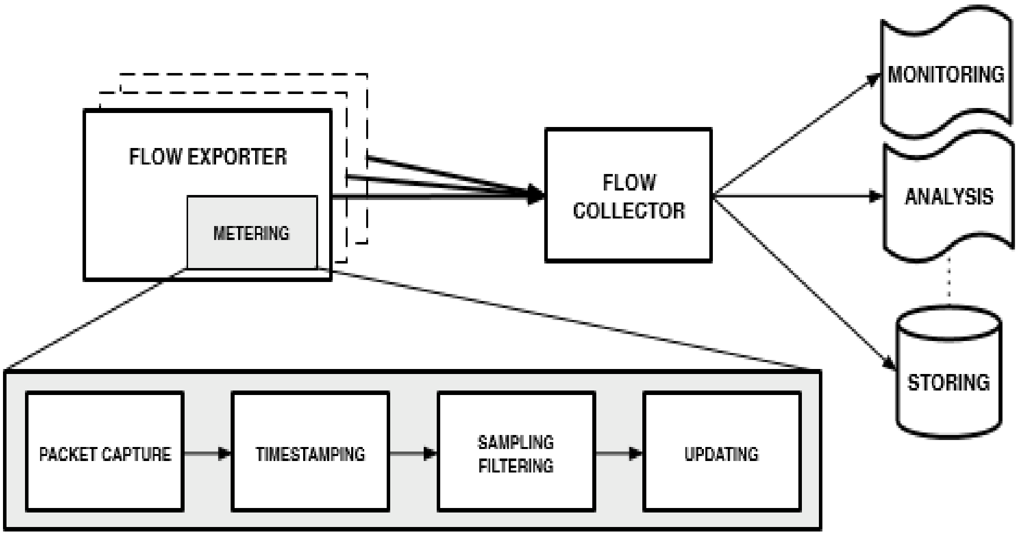


Fig. 1 IP flow exporting and collecting architecture [16], [7].

The final step is the update module. Each incoming packet header triggers an update to a flow entry in the flow cache. If there is no flow matching the packet header, a new flow entry is created. Once a flow record expires, it is sent to the flow collector.

## IV. FLOW-BASED SOLUTIONS FOR DOS

Detection of Denial of Service is often addressed in flow based intrusion detection. These attacks, by their nature, can produce variations in the traffic volume that are usually still visible at flow scale. It is important to underline that in case of flow-based detection we are implicitly addressing the problem of brute force DoS attacks, i.e., a type of DoS that relies on resource exhaustion or network overloading.

### *A. SYN Flooding*

A simple example of the use of sketches in DoS attacks is the detection of SYN Flooding attacks [12], as described in Gaoetal.[11].In this case, the sketch is supposed to store, for each time frame and each tuple (dest_IP, dest_port), the difference between the number of SYN packet sand the number of SYN/ACKs. If the stored value for the current time deviates from the expected one, a DoS SYN Flooding attack is going on

Another approach, in SYN flood attack, the victim sees a disproportionate number of SYN Packets compared to FIN packets. By a SYN packet, we mean any incoming packet with the SYN flag set. Recall that such a packet is a TCP connection request packet. Likewise, FIN and RST packets are, respectively, those with the FIN and RST flags set. A FIN packet is sent by the side that wishes to terminate the TCP connection. If the other party agrees to termination, it responds with its own FIN packet. Thus, SYN and FIN packets usually occur in pairs.

TCP connections that terminate normally involve one SYN packet (from the client) and a corresponding FIN packet to initiate or confirm termination of a connection. Thus the total number of incoming SYN packets should equal the number of incoming FIN packets.

Figure 2 shows two horizontal timelines- the top line shows the times of SYN packet arrivals and the bottom line shows the corresponding FIN arrivals. Time is slotted into fixed-length "Observation intervals", $T_1$, $T_{2...}$ during which we record the number of SYN arrivals. The corresponding observation intervals for $FIN_{S,}$ $T_1^{'}$, $T_2$,…are shifted to the right by the average duration of a TCP connection. To construct an anomaly detection system, we define the following variables as in [13].

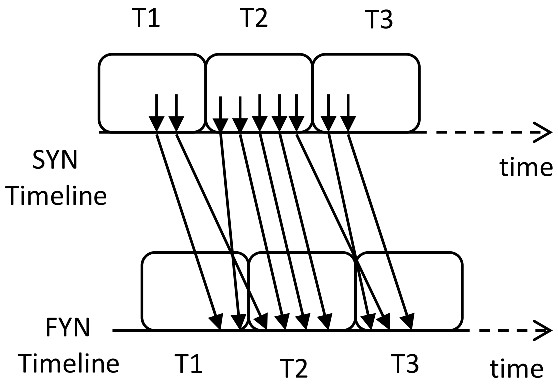


Fig. 2 TCP SYNs and matching FINs

Si = # of SYN packet arrivals in the i-th observation interval

Fi = # of FYN packet arrivals in the i-th observation interval

Di = normalized difference between # of SYN and FIN packets in the i-th observation interval, i.e.,

$$Di = \frac{Si - Fi}{Fi} \qquad (1)$$

T = threshold for detection

*Algorithm 1:* Raise an alarm if the most recently computed detection variable Di exceeds the threshold, i.e., Di>T.

### *B. SCAN*

Scans are usually characterized by small packets that probe the target systems. Keeping this characteristic in mind, it is easy to imagine that scans can easily create a large number of different flows.

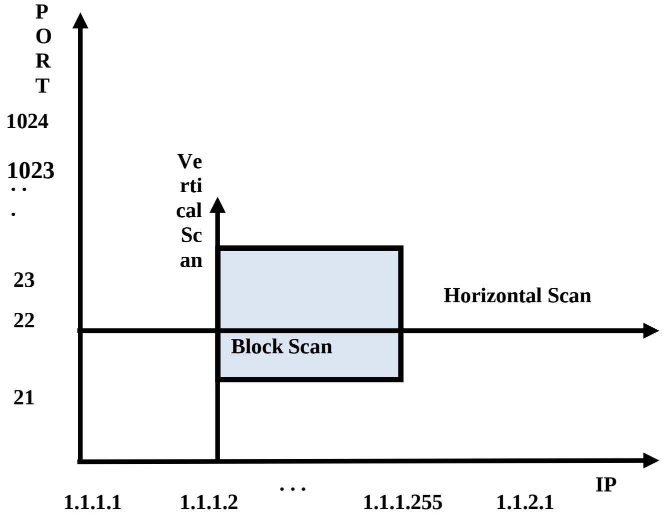


Fig. 3 Categories of scan [15]

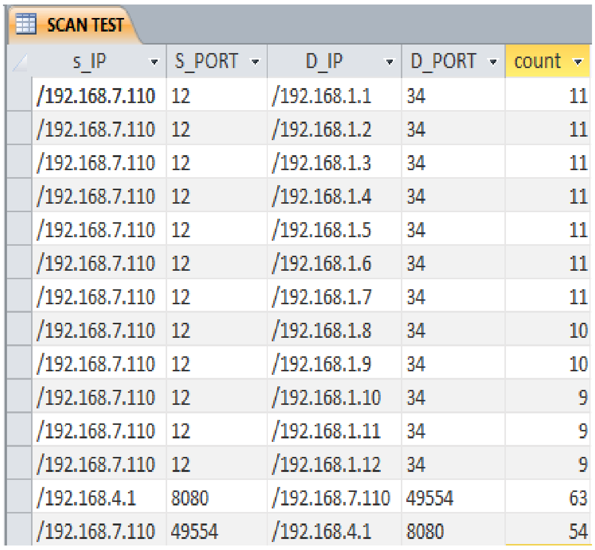
There are three categories of scans as shown in Figure 3: (i) a host scanning a specific port on many destination hosts (horizontal scan);(ii) a host scanning several ports on a single destination host (vertical scan); (iii) a combination of both (block scan). Irrespectively of the kind of scan, the result will be a variation of the flow traffic in the net work. At the same time, scans are less likely to have impact on the total traffic volume, as shown in Sperotto et al. [14].

## V. RESULT AND DISCUSSION

We applied and tested proposed method for our own different datasets having a suspicious flow. It is observed that the proposed system stores and updates the flow entries in database correctly. The snapshot of the database is shown in figure 4.

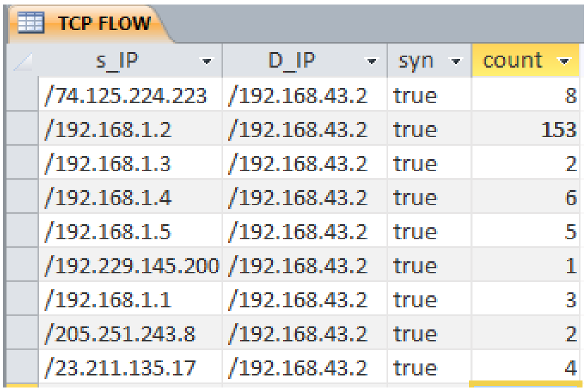
The host with IP address 192.168.7.110 has created 12 flow entries in the database as shown in Figure 4. All those flow entries consisting source as 192.168.7.110 and different destinations (192.168.1.1 to 192.168.1.12). The behaviour of source 192.168.7.110, shown in Figure 4, leads to horizontal scan for hosts in a particular network block.

Figure 5 shows the TCP flow flowing towards the host with IP 192.168.43.2 from different host in network. The flow entry with source IP 192.168.1.2 has SYN flow count of 153 which could be a suspicious flow because the



| s_IP | S_PORT | D_IP | D_PORT | count |
|---|---|---|---|---|
| /192.168.7.110 | 12 | /192.168.1.1 | 34 | 11 |
| /192.168.7.110 | 12 | /192.168.1.2 | 34 | 11 |
| /192.168.7.110 | 12 | /192.168.1.3 | 34 | 11 |
| /192.168.7.110 | 12 | /192.168.1.4 | 34 | 11 |
| /192.168.7.110 | 12 | /192.168.1.5 | 34 | 11 |
| /192.168.7.110 | 12 | /192.168.1.6 | 34 | 11 |
| /192.168.7.110 | 12 | /192.168.1.7 | 34 | 11 |
| /192.168.7.110 | 12 | /192.168.1.8 | 34 | 10 |
| /192.168.7.110 | 12 | /192.168.1.9 | 34 | 10 |
| /192.168.7.110 | 12 | /192.168.1.10 | 34 | 9 |
| /192.168.7.110 | 12 | /192.168.1.11 | 34 | 9 |
| /192.168.7.110 | 12 | /192.168.1.12 | 34 | 9 |
| /192.168.4.1 | 8080 | /192.168.7.110 | 49554 | 63 |
| /192.168.7.110 | 49554 | /192.168.4.1 | 8080 | 54 |

Fig. 4 Flow stored in SCAN Test



| s_IP | D_IP | syn | count |
|---|---|---|---|
| /74.125.224.223 | /192.168.43.2 | true | 8 |
| /192.168.1.2 | /192.168.43.2 | true | 153 |
| /192.168.1.3 | /192.168.43.2 | true | 2 |
| /192.168.1.4 | /192.168.43.2 | true | 6 |
| /192.168.1.5 | /192.168.43.2 | true | 5 |
| /192.229.145.200 | /192.168.43.2 | true | 1 |
| /192.168.1.1 | /192.168.43.2 | true | 3 |
| /205.251.243.8 | /192.168.43.2 | true | 2 |
| /23.211.135.17 | /192.168.43.2 | true | 4 |

Fig. 5 Flow stored in SYN Flood Test

matching FINs flow is absent for it.We can verify this by applying the algorithm 1 (see section IV) on above snapshot of flow, we get:

Di = (184-28)/28 = 6.57

To limit our scope, we will not consider how the threshold value $T_i$is decided. If we assume value of $T_i$ =4, the algorithm 1 will definitely raise an alarm. The drawback of algorithm 1 is, it may generate false positive and false negatives.

## VI. CONCLUSION

Since the flow-based intrusion detection system attracts the attention of researchers in last decade, it becomes the strong candidate for analysing the high speed networks. Also it reduces the data and processing time.

This paper shows how flow-based intrusion detection systems can be useful for detecting the DoS and scan attacks. Flow based techniques gives the better result to detect the DoS attack in high speed network as compared to payload-based techniques.

In some situations the complete absence of payload could be considered as the main drawback of flow-based approaches [1]. Flow-based intrusion detection cannot be considered as substitution for payload-based techniques, rather can be considered as complement for them in situations where detection must be done at high speed.

## ACKNOWLEDGEMENT

First of all, I would like to thanks my guide Dr. Sudhir S. Kanade, Head of Electronics Department, College Of Engineering Osmanabad, Maharashtra for guidance during my work, reading and comments on the manuscript of this paper. P.G. Coordinator Dr. Anil N. Holambe, for his valuable direction and guidance, Finally, I would like to thanks my wife for supporting me.

## REFERENCES

[1] Anna Sperotto, Gregor Schaffrath, Ramin Sadre, Cristian Morariu, Aiko Pras, Burkhard Stiller, "*An Overview of IP Flow-based Intrusion Detection*", IEEE Communications Surveys & Tutorials, Vol. 12, No. 3, Third quarter 2010.

[2] Computer Economics, "*2007 malware report: The economic impact of viruses, spyware, adware, botnets, and other malicious code*," Jul. 2008.[Online].

[3] M. Roesch, "*Snort, intrusion detection system*," Jul. 2008. [Online]. Available: http://www.snort.org

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

[5] H. Lai, S. Cai, H. Huang, J. Xie, and H. Li, "*A parallel intrusion detection system for high-speed networks*," in Proc. of the Second International Conference Applied Cryptography and Network Security (ACNS'04), pp. 439–451, May 2004.

[6] M. Gao, K. Zhang, and J. Lu, "*Efficient packet matching for gigabit network intrusion detection using TCAMs*," in Proc. of 20thInternational Conference on Advanced Information Networking and Applications (AINA'06), pp. 249–254, 2006.

[7] Cisco.com, "Cisco IOS Net Flow Configuration Guide, Release 12.4," http://www.cisco.com, Jul. 2008.

[8] B. Claise, "*Cisco Systems NetFlow Services Export Version 9*," RFC 3954 (Informational), Jul. 2008. [Online]. Available: http: //www.ietf.org/rfc/rfc3954.txt

[9] J. Quittek, T. Zseby, B. Claise, and S. Zander, "*Requirements for IP Flow Information Export (IPFIX)*," RFC 3917 (Informational), Jul. 2008. [Online].

[10] G. Schaffrath and B.Stiller, "*Conceptual integration of flow-based and packet-based network intrusion detection*," in Proc. Of2ndInternational Conference on Autonomous Infrastructure, Management and Security (AIMS '08), pp. 190–194, 2008.

[11] Y. Gao, Z. Li, and Y. Chen, "*A dos resilient flow-level intrusion detection approach for high-speed networks*," in Proc. of the 26th IEEE International Conference on Distributed Computing Systems (ICDCS '06), p. 39, 2006.

[12] S. M. Spechtand R. B. Lee, "*Distributed denial of service: Taxonomies of attacks, tools, and countermeasures*," in Proc. of the ISCA 17th International Conference on Parallel and Distributed Computing Systems (ISCA PDCS'04), pp. 543–550, Sep. 2004.

[13] Haining Wang, Danlu Zhang, Kang G. Shin, "*Change-poing monitoring for the detection of DoS attacks*", IEEE Trans. Dependable Sec. Comput., 1(4), pp. 193-208, 2004.

[14] A. Sperotto, R. Sadre, and A. Pras, "*Anomaly characterization in flow-based traffic time series*," in Proc.of the8th IEEE International Workshop on IP Operations and Management, IPOM 2008, Samos, Greece, pp. 15-27, Sep. 2008.

[15] J. Kinable. *Detection of network scan attacks using flow data*. In Proc. of the 8th Twente Student Conference on IT, 2008.

[16] B. Claise, "*Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of IP Traffic Flow Information*," RFC 5101 (Proposed Standard), Jul.2008. [Online].