International Journal of Computer and Electrical Engineering, Vol.2, No.6, December, 2010
1793-8163

Two-Stage Opportunistic Sampling for Network Anomaly Detection

Venkata Rama Prasad Vaddella, *Member IEEE* and Sridevi Rachakulla

***Abstract*—In this paper we propose the two stage opportunistic sampling technique for the detection and classification of network anomalies. Literature review indicates the application of one stage sampling for network anomaly detection. It is observed that for specific-purpose applications such as anomaly detection, a large fraction of information is contained in a small fraction of flows. We demonstrate that by using opportunistic and preferential sampling, the appearance and detection of anomalies within the sampled data set can be improved. We implement the two stage sampling and show that the results obtained are more effective. The evaluation of intelligent sampling techniques for improved anomaly detection is based on the application of an entropy-based technique on a packet trace. The proposed two-stage sampling reduces the time taken for the process when compared to the one stage sampling. We have also evaluated the results with different entropy values and observed the variation in flow distribution characteristics.**

***Index Terms*—Anomaly detection, Entropy, Intelligent sampling, Opportunistic sampling**

I. Introduction

Performance and traffic monitoring in computer networks has become essential for efficient management and reliable operation of networks. Due to the large number of flows on a high-capacity link it is very difficult to store and process all the flow information with a limited amount of resources. Thus, sampling has attracted a great deal of attention as a method to collect statistical information about flows. Studies have been made to analyze and evaluate the tradeoffs between sampling accuracy and efficiency, referring to the issues of minimizing information loss while reducing the volume of collected data. In this work we have made an attempt to use the effectiveness of two-stage sampling for network anomaly detection when compared to the one-stage sampling proposed in the literature [1].

Network anomaly detection is based on the concept that perturbations of normal behavior suggest the presence of anomalies (or attacks). The anomaly detection techniques rely on analysis of network traffic and characterization of the dynamic statistical properties of traffic [2]. Studies on the impact of random packet sampling on the blaster worm anomaly revealed that entropy-based metrics are less affected by sampling than volume-based metrics, and hence are more appropriate for anomaly detection purposes [3,4]. Based on the observation that for anomaly detection, a large fraction of information is contained in a small fraction of flows, we demonstrate that by using intelligent sampling techniques, we achieve "magnification" of the appearance of anomalies within the sampled data set by preferentially selecting appropriate data. In order to qualitatively and quantitatively evaluate the impact of intelligent sampling [5] effectiveness, an entropy-based anomaly detection method is proposed.

Manuscript received May 23, 2010. Revised August 20, 2010. This work is partially supported by the Management of Sree Vidyanikethan Engineering College under Annual Research Grant.

Dr. Venkata Rama Prasad Vaddella is working as Professor of Information Technology at Sree Vidyanikethan Engineering College, Tirupati, India (Corresponding author Phone: +91-877-2236711, Ext: 422, Fax: +91-877-2236717; e-mail: vvramaprasad@rediffmail.com).

Ms. Sridevi Rachakulla is at present working as a software engineer trainee at Tata Consultancy Services Limited, Hyderabad, India (e-mail: sridevi.rachakulla@gmail.com)

II. Network traffic Anomalies

We present three well-known malicious anomalies that could be characterized as network attacks; distributed denial of service (DDoS), worm propagation, and portscan. Two other common anomalies caused by legitimate network usage are: flash crowd and alpha flow. Figure 1 depicts these five anomalies in a network.

A. DDoS Attack

A Distributed Denial of Service (DDoS) attack [6] is characterized by an explicit attempt to prevent the legitimate use of service. DDoS attacks exploit known vulnerabilities of a communication protocol in order to disable the service requests. One common DDoS attack is SYNC flooding, where malicious sources send a large number of TCP SYNC packets to the victim's service, thus making the target machine unable to handle all these requests. Other types of DDoS attacks are UDP and ICMP flooding attacks where a high number of UDP or ICMP packets are sent toward the victim's network from multiple sources.

B. Worm propagation

The term *worm* [7] defines a malicious self-replicating program that tries to infect other machines by exploiting specific vulnerability. During the propagation phase, the infected machine sends a small number of packets per target to a large number of machines on the network.

C. Portscan activity

Portscan activity [8] includes traffic caused by a single machine that sends probe packets to a wide range of ports toward a specific host to check which services are available.

D. Flash crowd

A flash crowd event [9] consists of a large legitimate demand for a specific service (i.e. many clients simultaneously downloading the new release of a Linux distribution or a security patch from a HTTP/FTP server). This type of event results in the increase of both inbound

(requests) and outbound traffic (responses) from the HTTP/FTP server.

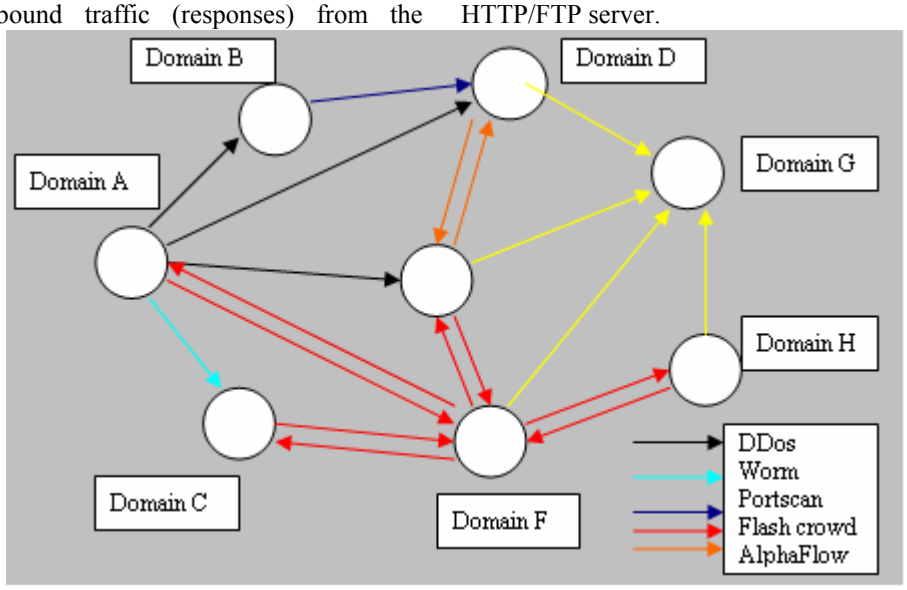

FIG.1 NETWORK TRAFFIC ANOMALIES

E. Alpha flows

Alpha flows [10] compose a network anomaly in which traffic increases from a few high-volume connections between two hosts. Alpha traffic is usually caused by large file transmissions over high-bandwidth links or network experiments between different domains.

III. TWO STAGE OPPORTUNISTIC SAMPLING

Sampling techniques can be divided in two major categories: packet-based and flow-based sampling. In packet-based sampling, packets are selected using either deterministic or nondeterministic method. In flow-based sampling, packets are first classified into flows. A flow is defined as a set of packets that have in common the packet header fields: source IP address, source port, destination IP address, destination port, and protocol. There are two stages for performing sampling: Sampling Stage and Prune Stage.

A. Sampling Stage

In this stage, sampling is performed in flows, which result in the selection of all packets that make up a particular flow. Application of packet sampling on network traffic measurements has been extensively studied in the literature [11,12], mainly for traffic analysis, planning, and management purposes. Recent results show that flow sampling improves estimation accuracy of flow statistics [11]. This fact makes flow sampling more suitable for anomaly detection purposes. Adaptive packet sampling techniques have also been studied [13]. We describe two well-known preferential flow-based sampling techniques. The first, called selective sampling [14], targets small flows (in terms of number of packets), while the second, called smart sampling [15], selects large flows. It has been demonstrated that small flows are usually the source of many network attacks (e.g., DDoS, portscans, and worm propagation) [2]; therefore, they should be preferentially selected in order to achieve high anomaly detection effectiveness. Selective sampling [14] follows this paradigm, and the selection of an individual flow is based on the expression:

$$p(x) = \begin{cases} c & x \le z \\ z/n. \cdot x & x > z \end{cases} \quad (1)$$

Where, x is the flow size in packets, $0 < c \le 1$, $n \le 1$ and z is a threshold (measured in packets).

From equation (1), it can be noted that flows that are smaller than z are sampled with a constant probability c, while flows that are larger in size than z are sampled with probability inversely proportional to their size. With an appropriate value for parameter c, a significant proportion of small flows can be selected without reducing the effectiveness of anomaly detection. Smart sampling [15] is a type of flow based sampling that focuses on the selection of large flows. In this, a flow of size x is selected with probability $p(x)$ according to the expression:

$$p(x) = \begin{cases} x/z & x > z \\ 1 & x \ge z \end{cases} \quad (2)$$

Where x is the flow size in bytes and z is a threshold.

In our work, we consider x as the flow size in packets. From equation 2 we observe that flows that are larger in size than z are sampled with probability 1, while flows that are smaller than z are sampled with probability proportional to their size. This sampling scheme is suitable for detecting anomalies caused by large flows such as flash crowd events and alpha flows.

B. Prune Stage

In this prune stage, pruning will be done on the already selected flows for anomaly detection process. Every time in the network there will be number of flows adding to it. The prune stage is introduced for every time-interval in the network. When the prune is completed, the sampling is repeated on the new flows. This will increase the efficiency of network anomaly detection process.

IV. ENTROPY

In this section, we present an entropy-based anomaly detection method that identifies network anomalies by

examining some characteristic traffic feature distributions. This method is independent of network topology and traffic characteristics, and can be applied to monitor any type of network. Entropy has been extensively used for anomaly detection purposes [16]. The entropy $H(X)$ of a data set $X = \{x_1, x_2 \ldots x_n\}$ is defined as,

$$H(X) = -\sum_{i=1}^{n} p_i \log_2 p_i \quad (3)$$

Where N is the number of elements contained in data set X and p_i is the probability $P[X = x_i]$. Entropy measures the randomness of a data set. High entropy values signify a more dispersed probability distribution, while low entropy values denote concentration of a distribution. Entropy values, as defined in Eq. 3, range between 0 and $\log_2 N$. We normalize the entropy by dividing $H(X)$ with the maximum entropy value $\log_2 N$. The normalized entropy is given by the following equation, whose the values range in (0,1).

$$H_n(x) = -\sum_{i=1}^{n} p_i \log_2 p_i / \log_2 N \quad (4)$$

Some common traffic feature distributions that are valuable in network anomaly detection are:

- The source IP address (srcIP)
- The destination IP address (dstIP)
- The source port (srcPort)
- The destination port (dstPort)
- The flow size (flow-size)

An anomaly such as an infected host that tries to infect other hosts in the Internet (worm propagation) results in decrease of the entropy of the source IP addresses. The infected machine produces a large number of flows, causing the same source IP address to dominate in the flow distribution of source IP addresses. On the other hand, during a port scanning activity, the entropy of the destination port increases due to the scan of random destination ports.

Based on these alterations, the network operator can identify the presence of an anomaly using predefined thresholds on the changes in the corresponding entropy values. Table-I summarizes the anomalies considered in this work and the corresponding changes in the traffic feature entropy values.

V. Distributed Denial of Service Attack

A. Sampling Stage

To emulate a DDoS attack we injected a proportion of SYNC flooding traffic within the actual operational background traffic. We consider 200 attackers sending TCP SYNC packets. Each of the attackers sends 20 flows consisting of one to four packets per time window. This causes a significant decrease in destination IP address and destination port entropy since the IP address of the victim and the specific target port (in our case TCP port 80) occur numerous times in a time window. On the contrary, entropy values for the source IP address and source port do not show significant alteration. In the first case, we choose $z = 4$, $c = 1.0$, and $n = 3$ to achieve the selection of all attack flows, and in the second case, $z = 8$, $c = 0.2$, and $n = 6$

TABLE I. CLASSIFICATION OF ANOMALIES BASED ON ENTROPY CHANGE

Anomaly	Description	Entropy change
Distributed denial of service (DDoS) attack	An attack on a specific service, making the resource unavailable to its users	Significant decrease in dstIP and dstPort. Almost no change in srcIP, srcPort, and flow-size.
Worm propagation	A self-replicating program that tries to infect other machines by exploiting a specific vulnerability	Significant decrease in srcIP and dstPort. Slight increase in dstIP and srcPort. Slight decrease in flow-size.
Portscan	Sending probe packets to a wide range of ports in a specific host to check which services are available	Significant decrease in srcIP, dstIP, and srcPort. Slight increase in dstPort. Slight decrease in flow-size.
Flash crowd	A large demand for a specific service (i.e., many clients downloading a specific file from an HTTP/FTP server)	Slight decrease in srcIP, dstIP, srcPort, dstPort, and flow-size.
Alpha flows	A small number of flows that have a very large quantity of packets (data transferred between two specific hosts)	Slight decrease in srcIP and dstIP. Almost no change in srcPort, dstPort, and flow-size.

.

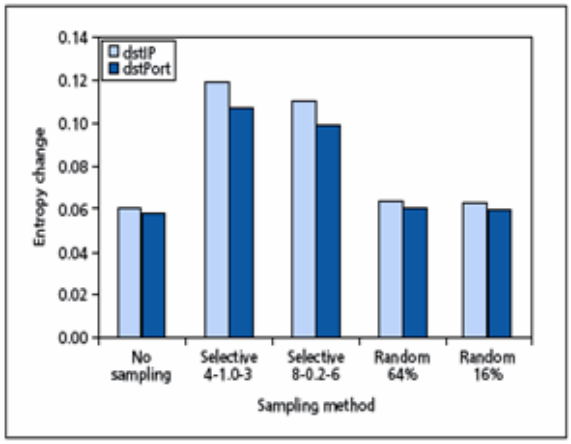

Fig. 2 Change in entropy due to DDoS attack

The behavior that shows entropy change effectiveness is attributed to the fact that in the first case of selective sampling all the attack flows have been selected while simultaneously a large proportion of normal large-sized flows has been discarded. In the second case, where $z = 8$, $c = 0.2$, and $n = 6$, the DDoS anomaly is again detected, but at a lower degree, as fewer attack flows have been selected.

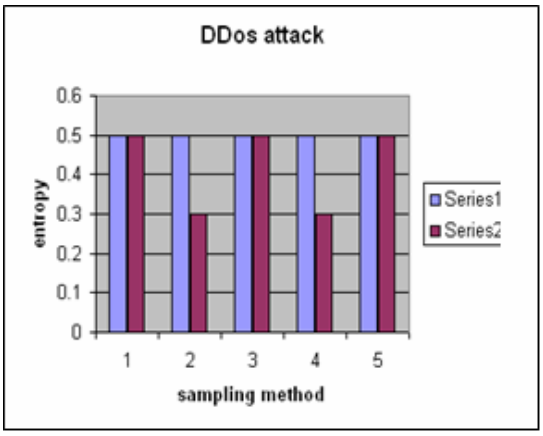

Fig. 3 DDoS attack with entropy ranging from 0.1- 1

Figures 2 and 3 show the change in source IP address and destination port entropy values between normal network operation and the anomaly period.

B. Prune Stage

In the pruning stage new flows are considered. The flow that contains the characteristics like similar source ip address, destination ip address are pruned for sampling. Only the flows that are not from same source and destination are considered for sampling and anomaly detection. Utilizing opportunistic sampling at the first stage, the final output would be a significantly reduced data set containing a great part of the anomalous traffic. The following tables show that every flow in the network initially checked for its existence. In Table-II all the flows in the network at a specified time window is considered. Each flow is recognized by its sourceIP address
and destIP address.

The already sampled field in the Table-II specifies that whether the flow has already been performed the anomaly detection or not. If there is any flows whose anomaly detection process has already been completed such flows will be pruned. The remaining flows now have to be sampled for anomaly detection. In Table-III, flow 2 is considered whose bandwidth x and threshold z is taken at the time of flow initialization, this helps to classify the type of flow (small flow- large flow). As z>x., the flow is small flow and selective sampling is performed. In Table-IV for the experimental purpose we have injected the DDoS attack. After injection we observe the change in destIP address and destPort address. Hence the DDoS attack is detected.

VI. Worm Propagation Case

A. Sampling Stage

A worm propagation scenario based on the slammer worm [17] is considered. The worm propagation phase includes only a single UDP packet targeted to port 1434 that correspond to 10 percent of the total number of flows in a time window. The worm propagation results in significant decrease in source IP address and destination port entropy, while the destination IP address and source port entropy slightly increase. We choose preferentially small flows, using the selective sampling method, as in the DDoS case.

TABLE II. PERFORMING PRUNING BY IDENTIFYING ALREADY SELECTED FLOWS

flow	source IP	destination IP	already sampled
0	134.123.125.101	134.123.125.105	TRUE
1	134.123.125.103	134.123.125.102	TRUE
2	134.123.125.109	134.123.125.109	FALSE

TABLE III. PERFORMING SAMPLING ON THE NEW FLOWS

flow	x	z	srcIPs	srcPorts	destIPs	dstPorts	type of sampling
2	4	5	4	4	4	4	Selective

TABLE IV. PERFORMING ANOMALY DETECTION ON THE NEW FLOWS

injected anomaly	srcIPs	srcPorts	destIPs	dstPorts	detected anomalies
DDoS	4	4	1	1	DDoS

Figures 4 and 5 show the change in source IP address and destination port entropy values between normal network operation and during the anomaly period.

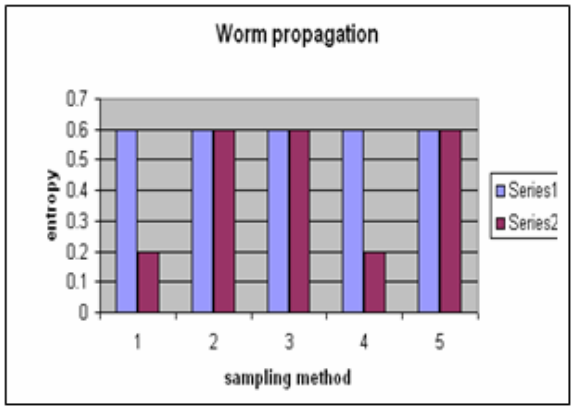

Fig. 4 Worm propagation attack with entropy ranging from 0.1-1

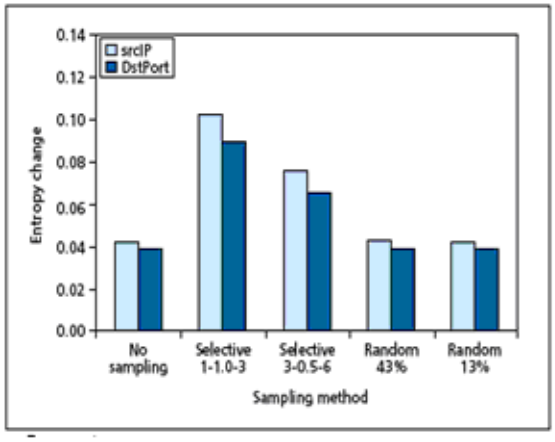

Fig. 5 Change in entropy due to worm propagation attack

B. Prune Stage

In the prune stage new flows are considered. The flow that contains characteristics like similar source IP address, destination IP address are pruned for sampling. Only the flows that are not from same source and destination are considered for sampling and anomaly detection. Utilizing opportunistic sampling at the first stage, the final output would be a significantly reduced data set containing a great part of the anomalous traffic. Tables V,VI, and VII show that every flow in the network initially checked for its existence.

In Table-V all the flows in the network at a specified time window are considered. Each flow is recognized by its SourceIP address and DestIP address. The already sampled field in the table V specifies whether the flow has already been performed the anomaly detection or not. If there are any flows whose anomaly detection process has already been completed such flows will be pruned. The remaining flows have to be sampled for anomaly detection.

In Table-VI flow 1 is considered whose bandwidth x and threshold z are taken at the time of flow initialization. This helps to classify the type of flow (small flow or large flow). As z>x, the flow is small flow, so selective sampling is performed. In Table-VII for the experimental purpose we have injected the worm attack. After injection we observe the change in sourceIP address and destPort address. Hence the worm attack is detected.

VI. PORTSCAN ACTIVITY CASE

A. Sampling Stage

In the portscan activity case we inject portscan traffic. A machine sends probe packets toward 1400 ports of the target machine and causes significant decrease in source IP address, destination IP address, and source port entropy. To enhance the detection effectiveness, we choose flows using the selective sampling method, as in the DDoS case. Figures 6 and 7 show the feature entropies that present significant change.

B. Prune Stage

In the prune stage new flows are considered. The flow that contains the flow distributed characteristics like similar source IP address, destination IP address are pruned for sampling. Only flows that are not from same source and destination are considered for sampling and anomaly detection. Utilizing opportunistic sampling at the first stage, the final output would be a significantly reduced data set containing a great part of the anomalous traffic. Tables VIII, IX and X show that every flow in the network is initially checked for its existence. If any flows that have already been in the network are pruned then new flows are selected for the process. In table VIII all the flows in the network at a specified time window are considered.

Table V. Performing pruning by identifying already selected flows

flow	source IP	destination IP	already sampled
0	134.123.125.102	134.123.125.105	TRUE
1	134.123.125.103	134.123.125.109	FALSE

Table VI. Performing sampling on the new flows

flow	x	z	srcIPs	srcPorts	destIPs	dstPorts	type of sampling
1	3	5	3	3	3	3	selective

Table VII. Performing anomaly detection on the new flows

injected anomaly	srcIPs	srcPorts	destIPs	dstPorts	detected anomalies
worm	1	3	3	1	worm

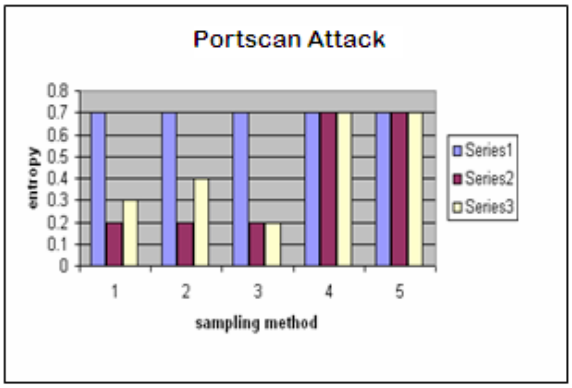

Fig. 6 Portscan attack with entropy ranging from 0.1-1

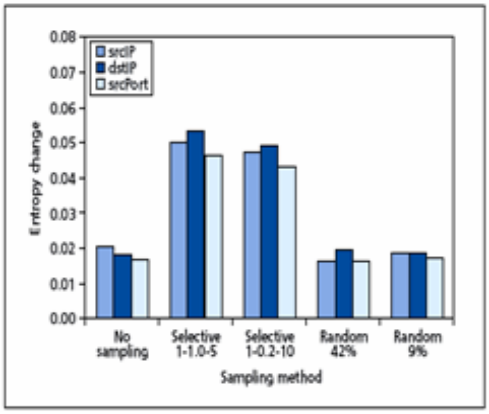

Fig. 7 Change in entropy due to portscan attack

Figures 6 and 7 show the change in source IP address and destination port and source port entropy values between normal network operation and during the anomaly period. Each flow is recognized by its sourceIP address and destIp address. The already sampled field in the Table-VIII specifies that whether the flow has already been performed the anomaly detection or not. If there are any flows whose anomaly detection process has already been completed such flows will be pruned. The remaining flows have now to be sampled for anomaly detection. In table IX, flow1 is considered whose bandwidth x and threshold z aretaken at the time of flow initialization, this helps to classify the type of flow (small flow or large flow). As z>x, the flow is small, so selective sampling is performed. In Table-X for experimental purpose we have injected the portscan attack, After injection we can observe the change in sourceIP address, sourcePort address and destIP address. Hence the portscan attack is detected.

VII. Flash Crowd Case

A. Sampling Stage

We study a flash crowd scenario in which 500 different clients from the internet requesting a specific file from a web server creates severe increase of flows in both directions of data transfers depicted in Table-I. The flash crowd event causes a decrease of all entropy metrics. We make use of the smart sampling method to select the appropriate data and utilize the following values for the parameter z. In the first case we choose $z = 400$, considering that the data transferred from the web server use flows of 600 packets. In the second case, where $z = 200$, we follow a more generic approach for targeting events that use large flows.

B. Prune Stage

In the prune stage new flows are considered. The flow that contains the flow distributed characteristics like similar source IP address, destination IP address are pruned for sampling. Only the flows that are not from same source and destination are considered for sampling and anomaly detection. Utilizing opportunistic sampling at the first stage, the final output would be a significantly reduced data set containing a great part of the anomalous traffic. Tables XI, XII, and XIII show that every flow in the network is initially checked for its existence. If any flow that has already been in the network is pruned, new flows are selected for the process. Figures 8 and 9 show the change in source IP address and source port and flow size values during normal network operation and the anomaly period.

Fig. 8 Flash crowd attack with entropy ranging from 0.1-1

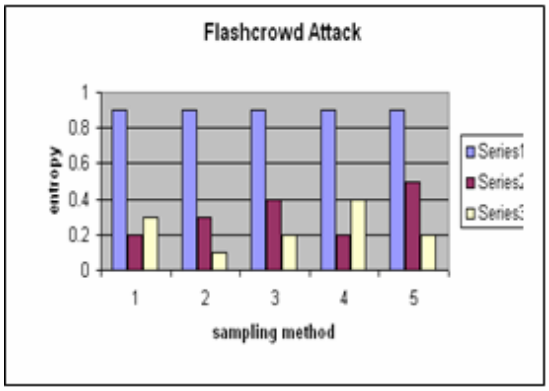

TABLE VIII. PERFORMING PRUNING BY IDENTIFYING ALREADY SELECTED FLOWS

flow	source IP	destination IP	already sampled
0	134.123.125.102	134.123.125.105	TRUE
1	134.123.125.103	134.123.125.109	FALSE

TABLE IX. PERFORMING SAMPLING ON THE NEW FLOWS

flow	x	z	srcIPs	srcPorts	destIPs	dstPorts	type of sampling
1	3	5	3	3	3	3	selective

TABLE X. PERFORMING ANOMALY DETECTION ON THE NEW FLOWS

injected anomaly	srcIPs	srcPorts	destIPs	dstPorts	detected anomalies

portscan	1	1	1	9	portscan

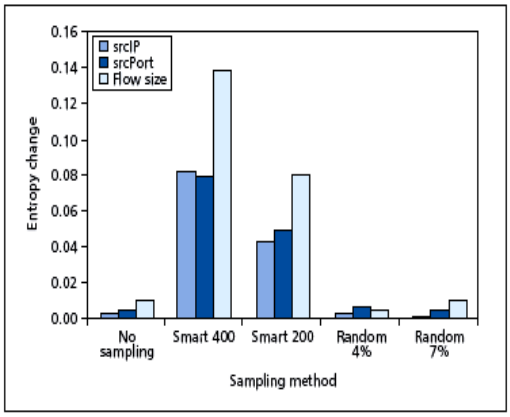

Fig. 9 Change in entropy due to flash crowd attack

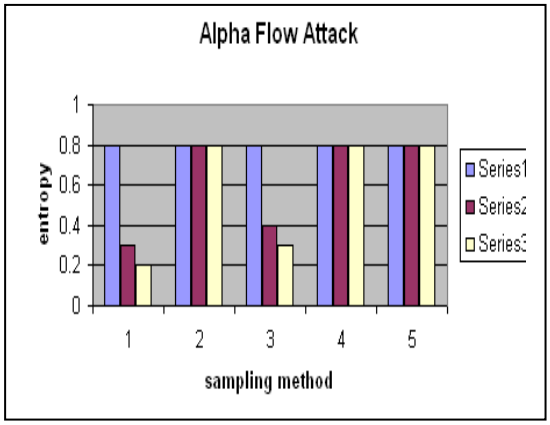

Fig. 10 Alpha flow attack with entropy ranging from 0.1-1

In Table XI all the flows in the network at a specified time window are considered. Each flow is recognized by its sourceIP address and destIP address. The already sampled field in the Table XI specifies whether the flow has already been performed the anomaly detection or not. If there are any flows whose anomaly detection process has already been completed such flows will be pruned. The remaining flows now have to be sampled for anomaly detection.

In Table XII, flow 0 is considered whose bandwidth x and threshold z are taken at the time of flow initialization. This helps to classify the type of flow (small flow- large flow). As z<x, the flow is large flow, so smart sampling is performed. In Table XIII for the experimental purpose we have injected the flash crowd attack and after injection we observe the change in sourceIP address, sourcePort address, destIP address and destPort address. Hence the flash crowd attack is detected.

IX. Alpha Flow Case

A. Sampling Stage

For this type of anomaly, we study a scenario in which two different hosts conduct huge data transfers comprising of 50 flows of 1000 packets each. As summarized in Table-I, alpha traffic causes a small decrease in the entropy values of both source and destination IP addresses. We chose values as per flash crowd case. We observe the change in source IP address, destination IP address, and flow-size entropy values between normal network operation and the period during the anomaly.

B. Prune Stage

In the prune stage the new flows are considered. The flow that contains the flow distributed characteristics like

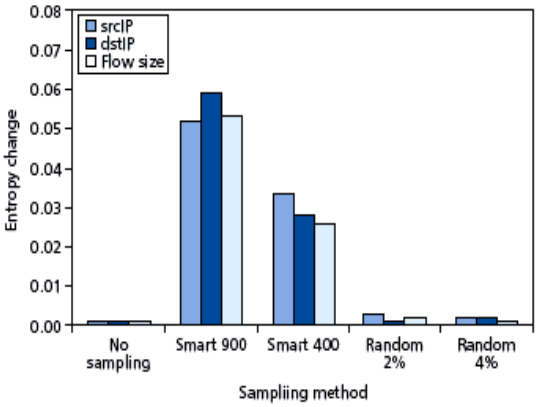

Fig. 11 1Change in entropy due to alpha flow attack

similar source IP address, destination IP address are pruned for sampling. The flow that contains the flow distributed characteristics like similar source IP address, destination IP address are pruned for sampling. Only the flows that are not from same source and destinations are considered for sampling and anomaly detection. To further reduce the amount of sampled data using the principle of two-stage sampling [18], where in the first stage sampling is performed at the flow level, and in the second stage packets are sampled from the already selected flows. Utilizing opportunistic sampling at the first stage, the final output would be a significantly reduced data set containing a great part of the anomalous traffic. Results in the Tables XIV, XV, XVI show that every flow in the network is initially checked for its existence. Figures 10 and 11 show the change in source IP address and destination port entropy values between normal network operation and during the anomaly period for an alpha flow attack.

TABLE XI. PERFORMING PRUNING BY IDENTIFYING ALREADY SELECTED FLOWS

flow	source IP	destination IP	already sampled
0	134.123.125.102	134.123.125.105	FALSE
1	134.123.125.103	134.123.125.109	TRUE

TABLE XII. PERFORMING SAMPLING ON THE NEW FLOWS

flow	x	z	srcIPs	srcPorts	destIPs	dstPorts	type of sampling
0	7	5	7	7	7	7	smart

TABLE XIII. PERFORMING ANOMALY DETECTION ON THE NEW FLOWS

injected anomaly	srcIPs	srcPorts	destIPs	dstPorts	detected

					anomalies
flashcrowd	1	1	1	1	flashcrowd

TABLE XIV. PERFORMING PRUNING BY IDENTIFYING ALREADY SELECTED FLOWS

flow	source IP	destination IP	already sampled
1	134.123.125.103	134.123.125.109	FALSE

TABLE XV. PERFORMING SAMPLING ON THE NEW FLOWS

flow	x	z	srcIPs	srcPorts	destIPs	dstPorts	type of sampling
1	8	5	8	8	8	8	smart

TABLE XVI. PERFORMING ANOMALY DETECTION ON THE NEW FLOWS

injected anomaly	srcIPs	srcPorts	destIPs	dstPorts	detected anomalies
slphaflow	1	8	1	8	slphaflow

In Table-XIV all the flows in the network at a specified time window are considered. Each flow is recognized by its sourceIP address and destIP address. The already sampled field in the Table-XIV specifies whether the flow has already been performed for anomaly detection or not. If there are any flows whose anomaly detection process has already been completed such flows will be pruned. The remaining flows have to be sampled for anomaly detection. In Table-XV flow 1 is considered whose bandwidth x and threshold z are taken at the time of flow initialization. This helps to classify the type of flow (small flow or large flow). As z<x, the flow is large flow, so smart sampling is performed. In Table-XVI for the experimental purpose we have injected the alpha flow attack. After injection we observe the change in sourceIP address and destIP address. Hence the alpha flow attack is detected.

X. TIME COMPARISON

Figure 12 shows the one-stage sampling and Figure 13 shows the two-stage opportunistic sampling. The graphs clearly indicate that with the increase of number of flows, the time taken to perform the anomaly detection process reduces in a two-stage sampling.

TABLE XVII. TIME COMPARISON OF ONE STAGE AND TWO-STAGE SAMPLING

Number of flows	Time (seconds)	
	One-Stage Sampling	Two-Stage Sampling
40	10	0.9
35	7.3	2
30	6	2.6
25	4.8	3
20	3	3.4
15	2.9	4
10	2	4.6
5	1	5

For example in a time window where 40 flows are present in the network, the time taken for anomaly detection is 0.9 seconds. This is due to the pruning of already detected flows for every time window in the network by the prune stage of the process. Only the fresh flows are given for detection process. This will save lot of time in a network. On the other hand, in a one-stage sampling, the time taken for the anomaly detection process with increase in number of flows in a network also increases. For example for 40 flows in a network for a time window it takes 10 seconds which is ten times more than the time taken to perform in the two-stage. This change is due to the fact that the here there is only a sampling stage but no pruning stage. Every flow is considered for anomaly detection.

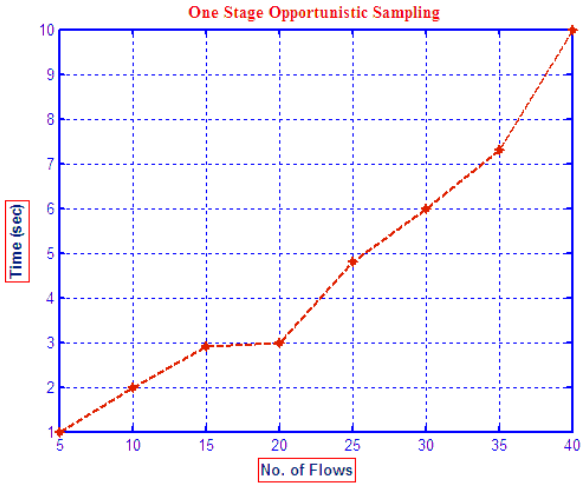

Fig. 12 One stage opportunistic sampling

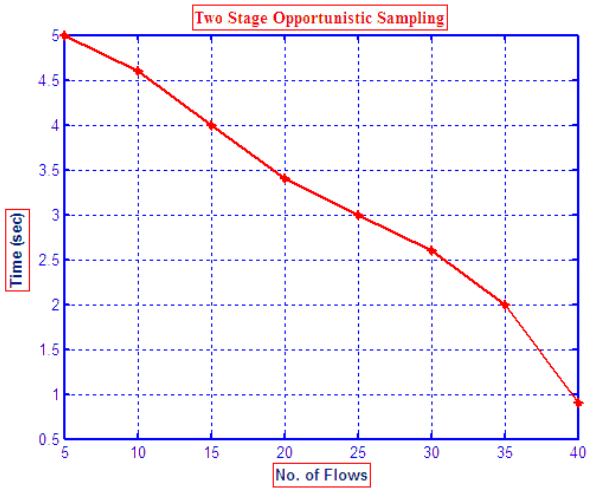

Fig. 13 Two stage opportunistic sampling

Finally, for a two-stage opportunistic sampling, as shown in the Table-VII, as the number of flows per time window increases, the time taken to perform the anomaly

detection process decreases. It clearly demonstrates the efficiency of the two-stage sampling when compared to the one stage sampling.

XI. Conclusions

In this work, the problem of improving network anomaly detection, effectiveness and classification through the application of two stage opportunistic flow sampling is discussed and experimental results are presented. The impact of two-stage flow based sampling technique on five different anomalies using an entropy based anomaly detection method has been evaluated, considering data that has been collected from an operational network. The experimental results demonstrate that even with small rates of anomalous traffic, intelligent sampling techniques significantly improve anomaly detection effectiveness and in several cases reveal anomalies that would otherwise be untraceable.

References

[1] Georgios Androulidakis, Vassilis Chatzigiannakis, and Symeon Papavassiliou, "Network Anomaly Detection and Classification via Opportunistic Sampling," IEEE Network, January-February 2009

[2] P. Barford and D. Plonka, "Characteristics of Network Traffic Flow Anomalies," *Proc. 1st ACM SIGCOMM Internet Measurement Workshop*, San Francisco, CA, Nov. 2001, pp. 69–74.

[3] J. Mai *et al*., "Impact of Packet Sampling on Portscan Detection," *IEEE JSAC*, vol. 24, no 12, 2006, pp. 2285–98.

[4] [4] J. Mai et al., "Is Sampled Data Sufficient for Anomaly Detection?," *Internet Measurement Conf. '06*, Rio de Janeiro, Brazil, Oct. 2006.

[5] D. Brauckhoff *et al*., "Impact of Packet Sampling on Anomaly Detection Metrics," *Internet Measurement Conf. '06*, Rio de Janeiro, Brazil, Oct. 2006.

[6] J. Mirkovic and P. Reiher, "A Taxonomy of DDoS Attack and DDoS Defense Mechanisms," *ACM SIGCOMM Computer Communication Review*, vol. 34, no. 2, Apr. 2004, pp. 39–53.

[7] S. Chen and S. Ranka, "Detecting Internet Worms at Early Stage," *IEEE* JSAC, vol. 23, No. 10, Oct. 2005, pp. 2003–12.

[8] A. Sridharan, T. Ye, and S. Bhattacharyya, "Connectionless Port Scan Detection on the Backbone," *Malware Workshop, IEEE IPCCC '06*, Phoenix, AZ, Apr. 2006.

[9] I. Ari et al., "Managing Flash Crowds on the Internet," *Proc. MASCOTS '03*, Orlando, FL, Oct. 2003, pp. 246–49.

[10] S. Sarvotham, R. Riedi, and R. Baraniuk, "Connection-Level Analysis and Modeling of Network Traffic," *Proc. 1st ACM SIGCOMM Workshop on Internet Measurement*, San Diego, CA, Nov. 2001.

[11] N. Hohn and D. Veitch, "Inverting Sampled Traffic," *Proc. 3rd ACM SIGCOMM Workshop Internet Measurement*, Miami, FL, 2003, pp. 222–33.

[12] N. Duffield, C. Lund, and M. Thorup, "Estimating Flow Distributions From Sampled Flow Statistics," *IEEE/ACM Trans. Networking,* vol. 13, no. 5, 2005, pp. 933–46.

[13] B.Y. Choi, J. Park, and Z.L. Zhang, "Adaptive Packet Sampling for Accurate and Scalable Flow Measurement," *IEEE GLOBECOM '04*, Dallas, TX, Nov.2004, pp. 1448–52

[14] G. Androulidakis and S. Papavassiliou, "Improving Network Anomaly Detection via Selective Flow-based Sampling," *IET Communication J*ournal, vol. 2, No. 3, Mar. 2008.

[15] N. G. Duffield and C. Lund, "Predicting Resource Usage and Estimation Accuracy in an IP Flow Measurement Collection Infrastructure," *ACM SIGCOMM Internet Measurement Conf. '03*, Miami Beach, FL, Oct. 27–29, 2003.

[16] S. Ranjan *et al*., "DoWitcher: Effective Worm Detection and Containment in the Internet Core," *IEEE INFOCOM '07*, Anchorage, AK, USA, May 2007, pp. 2541–45.

[17] D. Moore *et al*., "Inside the Slammer Worm," *IEEE Sec. & Privacy*, vol. 1, No. 4, July-Aug. 2003, pp. 33–39.

[18] L. Yang and G. Michailidis, "Sampled Based Estimation of Network Traffic Flow Characteristics," *IEEE I8FOCOM '07*, Anchorage, AK, USA, May 2007, pp. 1775–83.

Dr. Rama Prasad Vaddella, received the M.Sc(Tech.) degree in electronic instrumentation from Sri Venkateswara University, Tirupati in 1986 and M.E degree in Information Systems from BITS, Pilani, India in 1991. During the period 1989-1992 he worked as Assistant Lecturer in BITS, Pilani. From 1992 to 1995, he worked as Lecturer in Computer Science and Engineering and as Associate Professor from 1995 to 1998 at RVR & JC College of Engineering, Guntur, India. Since 1998, he is working as Professor and Head of Information Technology department at Sree Vidyanikethan Engineering College, Tirupati, India. He was awarded the Ph.D degree in Computer Science by J.N.T. Univeristy, Hyderabad, during 2007 for his thesis in *Fractal Image Compression*. He has also worked as a Research Assistant at Indian Institute of Science, Bangalore during the year 1986. He has published about 10 papers in national and international journals and presented several papers in National and International conferences. He has edited books, and refereed conferences. He is also a reviewer for 05 International Journals. His current research interests include computer graphics, image processing, computer networks, computer architecture and neural networks. He is a member of IEEE, ISTE and CSI.

Ms. Sridevi R received the B.Tech degree in Information Technology from J N T Univeristy, Anantapur, India during the year 2010. She has got selected as Trainee Software Engineer at Tata Consultancy Services, Hyderabad, India in April 2010. She was awarded first prize for the paper "*Digital Signature"* presented in the technical paper contest held at Loyola College, Chennai, India. She presented a paper "BLU RAY DISC" in the technical event conducted at V. R. Sidhartha college of Engineering, Vijayawada, India during March 2009. She is an IBM certified DB2 professional. Her current areas of research interest include Computer Networks, Software Engineering and Data mining. She is a member of ISTE.