*I. J. Computer Network and Information Security*, 2012, 9, 38-46
Published Online August 2012 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijcnis.2012.09.05

# Routers Sequential Comparing Two Sample Packets for Dropping Worms

Kannaiyaraja, Babu, Senthamaraiselvan, Arulandam
Manonmaniam Sundaranar University,S.K.P Engineering College, C Abdul Hakeem College of Engineering and Technology, Ganadipathy Tulasi's Jain College of Engineering
kanniya13@hotmail.co.in,babukannan5@gmail.com,senselvana@gmail.com,sakthisivamkva@gmail.com

***Abstract*** **—** Network IDS perform a vital role in protecting network connection in the worldwide from malicious attack. Nowadays the recent experiment work related to inspecting the packet for network security that is a minimal amount of process overhead. In this work, analysis the network intrusion for packet inspection that is together the testing data which inspect only group of packet selected as sample predominantly from small flows and select first two packets and comparing with each other overall packets and create tabelazied for find out different malicious debuggers. This experiment results shows that overcome the existing work .

***Index Terms*** **—** comparing packets, network intrusion detection system, probability of occurences, packet sampling method, router worms invention

## I. INTRODUCTION

Intrusion detection system perform has a device which acting as a software as application that monitor the protecting networks connected to the worldwide from malicious activities or violation of network. In this system sensor detects a potential security on networks which is a passive system, logs on the information and signal on alert to the network management. In reactive system which also known as intrusion prevention system (IPS) which is the auto system to response suspicious activities by resulting the connection. Conventionally IDS software product such as SNORT rule, Securenet, Hogwash work for monitoring traffic network on incoming IP packets is analysed for doughtful pattern which indicate antagonistic activities has these software system should compare packets against thousands of known patterns that must work with high accuracy at heavy traffic. The IDS is usually worked out to drop packets in the network and host based IDS solution are very difficult to control heavy traffic condition on monitoring the packets and then software solution was introduced which is also inherent limitation for network traffic control and finally introduced hardware solution which programmable gateway array which activate the same as IDS function with higher speed .But our research in the packet select randomly from the group of packets, compare each other, find out optimised worms or viruses, packet leak out from traffic network.

The purpose of this paper is to create innovative ideas of an analytical and statistical model of every IP packets monitories and inspected on the network flow. Recently Hepart and white have developed the first analytical work on the modelling of propagation of viruses it will be demonstration that markov chain management in multiple dimension which creates only on the previous packet in time, each packet in time, each packet depends on its neighbours packets in any of the multiple direction which comparing each other and make it visualization. Markov chain field is useful which implies that joint distribution at each packet in the network which could be computed in this manner.

The author who involved in internet and compute local area has decided the theoretical finding area. Additional technique of optimized SNORT rule implemented on the comparisons of packet with traffic generator and network analyser have further confirmed theoretical and experimental result to the structure and design of future ideas will be substantial one. The second session of testing the packet that historical model of network intrusion and third part we can create optimized network intrusion detection detection system, fourth part is the result are explored and fifth is the conclusion and future work.

## II. RELATED WORK

### *2.1) Evolution of Intrusion Detection:*

During the past five years, security of computer network has become main stream in most of everyone's life. Today most discussion on computer security is centred on the tools or techniques used in protecting and defending networks. The aim of this paper is to examine the origins of detecting, analysing and reporting of

malicious activity, where it is today and where it appears to be heading in the future. Some of the many techniques and tools presently used in network defence will be explored as well.

In fact, he was probably referring to the need of a risk assessment plan to understand the threat(what the risks are vulnerabilities, what the attacks might be or the means of penetrations) thus following with the creation of security policy to protect the systems in place.

Between 1984 and 1986, Dorothy Denning and Peter Neumann researched and developed the first model of a real time IDS.This prototype was named the intrusion detection Expert System (IDES)

The report published by James P.Anderson and the work on the IDES was the start of much of the research on IDS throughout the 1980s and 1995s. During this period, the U.S government funded most of this research.

To better understand the terms used within the ID user and research community, some of the most commonly used terms are:

a) *Host-Based:* The data from a single host is used to detect signs of intrusion as the packets enters or exists the host.

b) *Network-Based:* The data from a network is scrutinized against and it flags those who look suspicious. Audit data from one or several hosts may be used as well to detect signs of intrusions.

c) *Anomaly detection model:* The IDS has knowledge of normal behaviour so it searches for anomalous behaviour deviations from the established baseline. While anomaly detection's most apparent drawback is its high false positive, it does offer detections of unknown intrusions and new exploits.

In the last few years, the IDS field has grown considerably and therefore a large number of IDS have been developed to address specific needs. The initial ID systems were once anomaly detection tools but today, misuse detection tools dominate the market. With an increasingly growing number of computer system connected to networks, ID has become a necessity. In the mid 1165s, commercial products surfaced for the masses.

Two of the most popular IDS in the mid 1995s were Wheelgroup's Netranger and Internet Security System's Realsecure. Both of these companies started out with network- base IDS.

Wheelgroup was formed in October 1995 to commercialize a security product initially prototyped by the U.S Air Force then called Netranger. This product "scans traffic for "signature of misuse", providing real-time alarm and details of the furtive attacks that may fingerprint=AF19 in February 1998, wheel group was 06E4 A169 4E46 key plague a network", 5FA27 2F94 188D FDB5 DE9D F8B5 acquired by Cisco to eventually become an integral part of Cisco's security architecture.

Internet Security Systems, Inc (ISS) was founded in April 1994 by Thomas Noonan and Christopher Klauss, after Mr.Klauss invented and released the first version of the Internet Scanner. On 9th December 1996, ISS announced the release of a tool to augment network security with real-time attack recognition called Real Secure. On the 19$^{th}$ August 1997, they announced the first commercial released of their IDS called Real Secure 1.0 for Windows NT 4.0 a new commercial breakthrough.

Another point to consider is most commercially available systems are knowledge-based, which means matching signatures of known attacks against changes in systems or streams of packets on the network. However, their major weaknesses are they often helpless against new attacks, so they must be continually updated with new knowledge for new attacks signatures. Despite the fact these false positives are common with behaviour-based IDS, so is its ability to detect a previously unreported attack.

*2.2) Analysis model of Network Intrusion:*

The analysis that will be now developed based on selecting the packet as sampling that was advocated by markov process chain model. The markov process chain model is analysed to the process of network intrusion, the related formulas the probability of network being compromised. It will be demonstrated that it is possible to select inspect the packets arriving into the network while maintaining the high degree of security and performed such inspection with related formula. Consider an intranet that is connected to the worldwide which protected by IDS shown in the figure1.

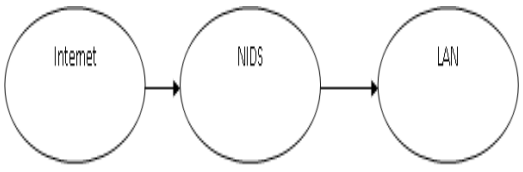



Figure 1-Network Intrusion Model

We could assume that at any given intranet has total of n process in which i$^{th}$ process may be intrusive. All the characteristics in the network such $K_i$ when ith intrusive process are running, state $S_{i-1}$ which also running etc. which are chain state management which have no intrusive activities inside it. The first foremost step

towards the process of intrusion is to realize that the different states of network can be regarded as set of mutually exclusive and collectively possible states. Furthermore the transition of the network in the presence state which are performed to different state and find out the probability of transition to the next states. By using Marko process of network intrusion modelled after popular birth and death [1, 2] epidemiological model indeed. This model is based on that assumed to make a transition from state $S_{i-1}$ to $S_i$. The probability of transisition in this model $S_i$ -> $S_{i-1}$ is not equal to the probability of transition.

## III. CONSTRUCTION

### *3.1) Method of selective packet inspection:*

We suggested the formula the following for full inspections of all the packets should be implemented by the value of $P_{th}$ is calculated threshold and selective inspection possible $P_{th.}$ Under the heavy traffic congestion and low probability of occurrences of intrusion such a solution is explicitly required we could now answer the important questions of what kind of packet might be used that ensure to the intrusion would be detected.

Consider an internet that is connected to the World Wide Web and protected by network intrusion detection system as shown in the figure1.we shall create that the internet has a total of N process of which $i^{th}$ process may be intrusive or hostile. We shall develop the network has been the states $S_i$ when I hostile process are running, state $_{i-1}$ also running while $s_0$ will be a clean state that means there is no intrusive activities exists the first steps of modelling is the process of intrusion which realized the different states of network can be recorded as a set of mutually exclusive state and also the transition of network from the current state to the different state is a function and clarify the probability of transition to the next state. These characteristics are the markov process chain model [9][4].

Wang and Wang [5] are emphasizing the process of network intrusion by using popular birth and death model has been used in a number of other engineering problems. This model is based on assumption is equally for a system to create a transition state $S_{i-1}$ to state $S_i$ and state $S_i$->$S_{i-1}$ for understanding this impact. In figure2 $i^{th}$ hostile processes which are running on the network may be initiated new hostile process. Therefore the number of process become i+1, if create the termination on the ith process which is i-1 and new hostile process to be started after initiated existing hostile process to be terminated that is named as birth and death model. This model will be now used in the present analysis such as $S_i$, $S_{i-1}$.In this analysis presented here will not be altered but additional constant of simply appear in the final equation.

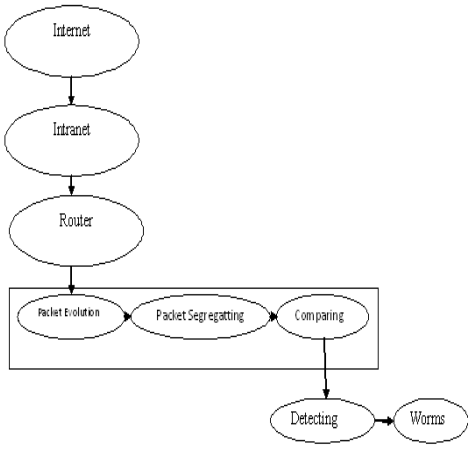



Figure 2-Packet Flowing Structure Architecture

We now define four variables for different states in the figure2.Such as internet, routers, packet evolution, packet segregation, packet comparison are forced to find out worms create in the network packets and the routers which receiving packet from other sources which are to be selected $S_1$->$S_{i-1}$ and $S_2$->$S_{i-2}$ which are analysed by packet evolution for segregating the packets. The router basic assumption that specifically described four variables let us, b be birth rate and d be the death rate and bi has consider as probability and $P_{th}$ the probability that any process started on the network be a hostile process.

$$\text{First Packet}=1/n\ dp1=bp_{th}\ p_0 \quad \rightarrow 1$$

Above this formula are used in our analysis presented selecting the first packet and compare with one another by using the following assumed formula that

$$\text{Second Packet}=2/n\ dp_2 =bp_{th}\ p_1 \quad \rightarrow 2$$

The number of processes initiated by users and does not include such things as system processes or background tasks, since the probability of occurrence of intrusion is associated with user processes only. For the n = 100 curve, for example, it can be easily concluded that the inspection of all the incoming packets will be inevitable if the probability of occurrence of intrusion is larger than about 2%, as the probability of a clean state is practically zero at all points past that threshold. For PH < 2%, the probability of a clean state is substantial, and, as will be demonstrated in the following section, it is possible to inspect packets selectively under such conditions without sacrificing security. This is clearly a better alternative to the strategy of "inspect all packets, or drop packets randomly" that is currently being implemented in IDS software solutions.

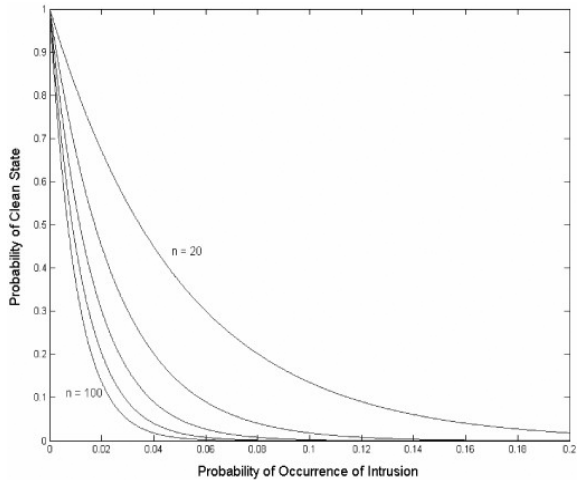
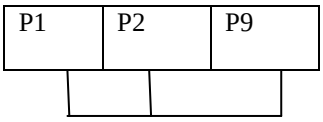



Figure 3- Probability that network is in clean state (P0) versus the probability of occurrence of intrusion (PH), for values of n ranging from 20 to 100.

And comparing each other and find out worms occurred frequently by using algorithm.

*Router worms Invention Algorithm:*




Comparison of packets

Constant p [ ] = {packet mapped information as flag}

If p<r then

Selection (p,i,j,r)

Comparison (p,i,r-1)

return p;

Comparison (p, i, j, r)

x<- p[r], i<- p1

w<- worms

for (j<-p to r-1)

j=j+1

do if p[j] <p[j+1]

find (worms in the packet)

then i<-i+1

tabalizied (p,i,j,r)

return i+1;

*3.2) Packet Sampling Method*

In this Section, we introduce the ideal flow sampling process, which is used for selecting the most relevant packet flows .Intuitively, the ideal sampling should be a process in which number of samples as selected from first two packets and selected packet has been inspected by Router worms Invention Algorithm and tablazied the worms created from the packet. The most worms are frequently occurred are noted and find out. Almost all existing anomaly detection methods use statistical distribution of flows to model the network traffic. That means to minimize the loss of information, it is reasonable to preserve as much of the statistics as possible.

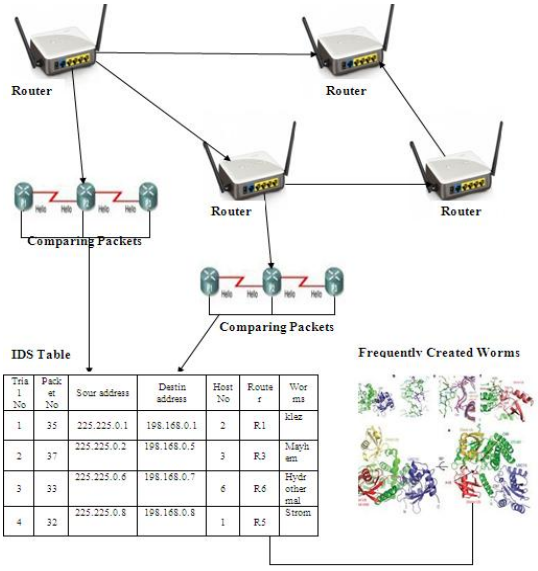



Figure 4-Packet sampling model architecture

Each flow x=1 can be identified by a set of features like source IP address or protocol. We will denote $k_{th}$ feature as Xk and the probability of selecting a flow x=1 into the sampled set as p(x=1). Furthermore, the statistical information is captured by feature moments which are computed from feature values. We distinguish between two types of feature moments:

- Feature counts c((x=1) jXk) indicating summations and numbers of flows related to x through the feature Xk (in flows, packets or bytes). We will implicitly use these feature counts in number of flows unless told otherwise.
- Feature entropies eXk ((x=1)jXl) describing the entropy of feature Xk of flows related to x through the feature Xl.

Finally we will denote the original finite unsampled set as U and the finite set of samples as S. Thus cS(xjsrcIP) denotes the number of flows from the sampled set with exactly the same source IP as has this flow x. While eU

sIP (xjdP) is the entropy of source IP addresses from the original unsampled set, whose flows target exactly the same destination port as flow x. When we will consider feature counts across more (q) features, we will denote them as c(xjX1,….,Xq) etc.

By using feature moments, it is possible to compute various Net Flow traffic characteristics used in vast majority of anomaly detection methods. The loss of information affects these moments by shifting their original values, which has negative impact on the consequent detection performance. Therefore the ideal sampling should minimize the loss of information of these moments.

*Definition 1:* Let S1,…., Sm is various sets of flows selected from U with probability p(x). Feature moment c(xjXk) is reversible in U if and only if:

Packet (x=1)

$$\forall x \in \mathsf{U}: \lim_{m\to\infty} \sum_{i=1}^{m} \left( \mathsf{c}^{\mathsf{U}}(x|X_k) \cdot p(x) - \mathsf{c}^{\mathsf{S}_i}(x|X_k) \right) = 0 \quad \rightarrow \quad 3$$

Feature reversibility ensures that the loss of information is minimal, allowing computation of the original (unsampled) moment value.

To define the reversibility of entropy feature moments, it is reasonable to use relative uncertainty instead of entropy values, because relative uncertainty better specifies feature distributions. We will denote relative uncertainty that describes normalized entropy feature moment eXk (xjXl) as:

$$RU(\mathsf{e}_{X_k}(x|X_l)) = \frac{\mathsf{e}_{X_k}(x|X_l)}{\log \mathsf{c}(x|X_l)} \in [0,1] \quad \rightarrow \quad 4$$

*Definition 2:* Let S1,….,Sm be various sets of flows selected from U by using probability p(x). Feature moment eXk (xjXl) is reversible in U if and only if:

$$\forall x \in \mathsf{U}: \lim_{m\to\infty} \sum_{i=1}^{m} \left( RU(\mathsf{e}^{\mathsf{U}}_{X_k}(x|X_l)) - RU(\mathsf{e}^{\mathsf{S}_i}_{X_k}(x|X_l)) \right) = 0 \quad \rightarrow \quad 5$$

*Definition 3:* Let Xi be i-th flow feature. Feature variability V(XUi ) of feature Xi is defined as the number of distinct values of feature Xi in U.

*Definition 4:* Ideal sampling with sampling probability p(x) is defined as a sampling where:

1) All feature moments (counts and entropies) are reversible
2) Coefficient $V(X_t^s)/V(X_t^u)$ for all features is maximized.

Each of the criteria caters to different kind of anomaly detection approaches: feature moment reversibility is essential for the methods based on statistical and pattern recognition methods, while the feature variability is also essential for knowledge-based approaches that depend on specific values of individual features. This idealistic process defines two actually usable quality metrics, which can be applied to any implemented sampling method in order to quantify the quality of the result it provides from the anomaly detection standpoint:

*1) Feature representation* – describes the deviation (interval [0,1]) of a probability distribution from the ideal distribution, and thus measures the reconstruction error in the reversibility of count moment c(xjXk):

$$f_{\mathsf{c}}^{rep}(X_k) = \frac{1}{|\mathsf{U}|} \cdot \sum_{\forall x \in \mathsf{U}} | \mathsf{c}^{\mathsf{S}}(x|X_k) - p(x) \cdot \mathsf{c}^{\mathsf{U}}(x|X_k) | \quad \rightarrow \quad 6$$

and the reconstruction error of entropy moment eXk (xjXl):

$$f_{\mathsf{e}}^{rep}(X_k, X_l) = \frac{1}{|\mathsf{U}|} \cdot \sum_{\forall x \in \mathsf{U}} | RU(\mathsf{e}^{\mathsf{U}}_{X_k}(x|X_l)) - RU(\mathsf{e}^{\mathsf{S}}_{X_k}(x|X_l)) | \quad \rightarrow \quad 7$$

*2) Feature coverage* – describes the variability of feature:

$$f^{cov}(X_i) = \frac{\mathrm{V}(X_i^{\mathsf{S}})}{\mathrm{V}(X_i^{\mathsf{U}})} \in [0,1] \quad \rightarrow \quad 8$$

We use these measures to compare the properties of proposed sampling technique with two existing sampling techniques (random and selective [9]).

*3.3) Optimization of network intrusion detection systems*

In view of the above, two questions must now be answered: first, how the selected first pa in the network by using first serve memories ted the sampling, secondly the same kind of packet are selected sampling strategy used to ensure that intrusion would still be detected if it occurs. In 2008, Androulidakis and Papavassiliou [6,8] demonstrated experimentally for the first time that under certain conditions, the selective inspection of packets for the purpose of detecting network intrusion can be as effective as the full inspection of all packets. We shall now demonstrate that the Androulidakis-Papavassiliou criterion corresponds with the conclusions reached above.

*A) The connection between PH and the Androulidakis-Papavassiliou criterion*

Androulidakis and Papavassiliou suggested that by selectively inspecting packets and calculating the Shannon entropy for the packets selected, a number that is indicative of the likelihood of occurrence of intrusion is obtained. The Shannon entropy H is defined as

$$H = \sum_{k=1}^{N} P_k \log_2 \frac{1}{P_k} \rightarrow \quad 9$$

where N is the number of packets inspected and Pk is the probability of occurrence of message k within the stream of packets selected. The "messages" of concern here are the following: the source IP address, the destination IP address, the source port, the destination port, and the protocol. According to [8] "Entropy measures the randomness of a data set. High entropy values signify a dispersed probability density function, while low entropy values indicate a more concentrated distribution. For example, an anomaly such as an infected host on the Internet that tries to infect other hosts (worm propagation) results in a decrease of the entropy of the source IP address. The infected machine produces a disproportionately large number of packets, causing the same source IP address to dominate in the distribution of source IP addresses". The entropy according to Eq. (9) has a maximum value of log2 N. In [8], the entropy was "normalized" by dividing the expression by log2 N, so that it ranges from 0 to 1, that is

$$H(\text{normalized}) = \frac{\sum_{k=1}^{N} P_k \log_2 1/P_k}{\log_2 N} \rightarrow \quad 10$$

While the normalized entropy takes values in the range (0, 1), it is actually an inverse measure of the probability of occurrence of intrusion. As indicated above, an intrusion attempt would actually cause the value of the entropy of the source IP address to decrease. A value of H that is close to 0 indicates a probability of occurrence of intrusion that is close to 1 and vice versa. Hence, the probability of occurrence of intrusion PH can be defined in terms of H as follows:

$$P_H = 1 - H(\text{normalized}) = 1 - \frac{\sum_{k=1}^{N} P_k \log_2 1/P_k}{\log_2 N} \rightarrow \quad 11$$

To test the theoretical predictions made in this paper a number of tests that involved the network of a local corporation were conducted. The first test was a test to determine whether PH as defined by Eq. (11). In the test, the LAN was subjected to traffic that emulates the propagation of the Slammer worm [7] (additional details about the tests are discussed in the following sections). The malicious traffic consisted of a single UDP packet per destination IP address, where the destination IP address was chosen randomly. The source port of each UDP packet was also chosen randomly, ranging from 1 to 65,595. The packets arriving at the LAN were inspected selectively, and the value of PH was calculated from Eq. (11) at regular time intervals. The number of user processes n running on the LAN was purposely maintained at a constant value of approximately 50. The result is shown in Figure 9. As can be seen from the graph, the calculated value of PH increased from less than 1% to more than 15% for time duration of approximately 1 min during which the attack was simulated. This corresponds very well with the data shown in Figure 3. Clearly, the formula in Eq. (11) for calculating the instantaneous value of PH correlates with the analysis of the previous section.

*B) The principle of selective packet inspection*

As suggested by algorithm and figure 3, full inspections of all the packets should be implemented by the IDS if the value of $P_H$ is calculated first two packets arriving in the router, and selective inspection is conceivably possible. Under heavy traffic conditions and low probability of occurrence of intrusion, such a solution is obviously very desirable.

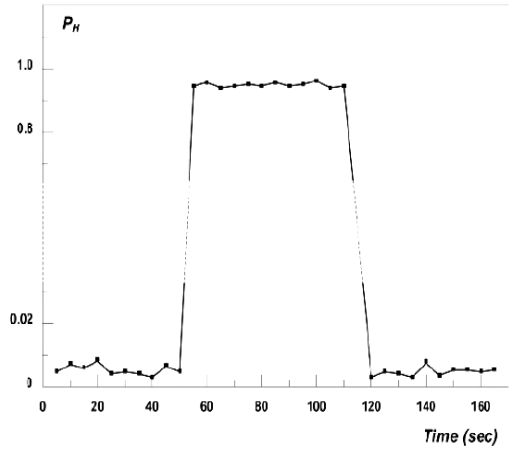



Figure 5 -Value of PH as calculated from Eq. (11) for the source IP address and/or the destination port (worm propagation test).

We shall now answer the important question of what kind of packet sampling strategy must be used to ensure that intrusion would still be detected if it occurs. A number of studies have differentiated between packet-based sampling and flow based sampling [11-12]. In packet-based sampling, packets are selected from the global traffic using a pre-specified method. In flow-based sampling, packets are first classified into flows. A "flow" is defined as a set of two packets that have in common the following packet header fields: source IP address, destination IP address, source port, destination port, and protocol. They have showed that small flows (flows that consist of 1-4 packets) are usually the source of most network attacks. Androulidakis and Papavassiliou have in fact advocated and demonstrated the success of the selective inspection of packets from small flows in their experimental investigation. According to that approach, flows that consist of 1-4 packets are fully inspected, and larger flows are inspected with a sampling frequency that is inversely proportional to their size (see ref. [6]). We now give a rigorous proof that such a technique for the selective inspection of packets guarantees that intrusion will be detected if it occurs:

*Lemma:* If the selective inspection of packets with a sampling probability that favors small flows is implemented, the probability of detecting a network intrusion is approximately equal to 1.

*Proof:* Assume that the probability of any packet selected being a malicious packet is P. If a total number of N packets are selected, the probability of detecting at least one malicious packet will be given by the Bernoulli statistical trial probability [15]

$$P_{\text{detection}} = \sum_{r=1}^{N} \binom{N}{r} (P)^r (1-P)^{N-r} \quad \rightarrow \quad 12$$

For sufficiently large N (e.g., N > 50) and sufficiently large P (e.g., P > 0.01, or 1%), the above summation is approximately equal to 1. If packets are selected predominantly from small flows, P is guaranteed to be substantially higher than 1% (port scan, for instance, is only one packet).

To summarize the above conclusions, modern, efficient IDS should selectively inspect packets such that small flows (flows that consist of 1-4 packets) are fully inspected, and larger flows are inspected with a frequency that is inversely proportional to their size. The probability of occurrence of intrusion PH should be calculated in real time by using Eq. (11). For calculating PH, only the packet headers need to be inspected (see the discussion in the previous section) and the probabilities of occurrence of the source/destination IP address, the source/destination port, and/or the protocol must be calculated and used in Eq. (11). If at any time PH exceeds a suitable threshold that is calculated from Eq.(11), the IDS must switch immediately to the full inspection of the content of all the packet traffic and quarantine any packets that are found to be malicious.

### *C) Testing of the proposed IDS approach*

The local area network of a small local corporation of 50 employees was used to test the IDS approach suggested above. The experimental setup is shown in Figure 6. As shown, malicious traffic was generated from a Linux machine on which two different packet-generation programs were installed: IDSWakeup [16] and DITG [17]. These programs make use of the powerful kernel of Linux to generate packets at speeds of up to one Gigabit per second. The main purpose of IDSwakeup is to generate false intrusive attacks that mimic well-known ones (e.g., Denial of Service (DoS) attacks, port scan, and worm propagation), in order to determine how the IDS detects and responds to those attacks. D-ITG (which stands for Distributed Internet Traffic Generator), on the other hand, is a simple but very versatile packet generator that can generate packets of different sizes and different inter-departure times. The malicious traffic generated was merged with regular Internet traffic through a Cisco router and directed to the corporate LAN, as shown. A simple IDS software solution was developed for implementing the inspection strategy described above. The code was developed in Matlab and converted to C (for brevity, the details of the code will not be discussed here). Essentially, the code inspects the headers of the packets in small flows . The headers of packets in larger flows are inspected with a frequency that is inversely proportional to the size of the flow, as described in the previous section. After 50 packets are selected, the code computes PH from Eq. (11), for 9 different attack scenarios: DoS, port scan, and worm propagation. If PH is found to have exceeded a suitable threshold that is calculated by using algorithm, the code immediately moves to full inspection mode, where the actual contents of the packets selected and all subsequent packets are inspected for the presence of well-known patterns [11,28,29]. Any packets that are found to be malicious are quarantined. Throughout each test conducted, the number of user processes n running on the LAN was purposely maintained at a constant value.

The first objective of the testing was to determine the number of malicious packets that managed to slip through the IDS when PH was below the calculated threshold.

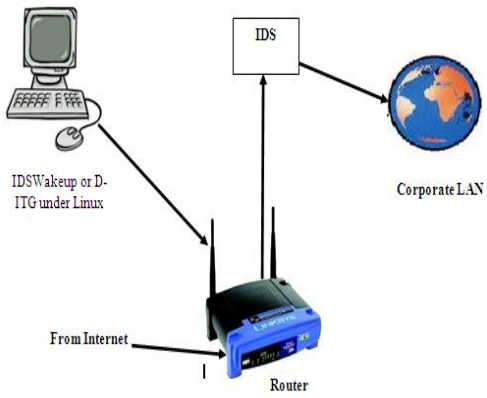



Figure 6- Setup for testing the proposed optimized IDS approach.

Figures 7 and 8 show the results that were obtained for an average number of user processes n = 1 and 50, respectively. As Figure 7 shows, the maximum percentage of hostile packets that slipped through the IDS in the first case was slightly over 0.1% .The results were very similar for the 9 types of attacks: DoS, port scan, and worm propagation. In this test, an extremely small percentage of the global flow was made hostile, instead of actually launching an outright intrusive attack. This percentage was then increased gradually, which helped increase the calculated value of PH, as the graph shows.

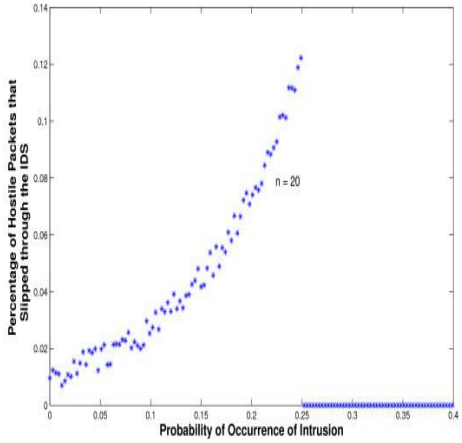



Figure 7- The percentage of hostile packets that "slipped" through the IDS as a function of PH, for n = 1.

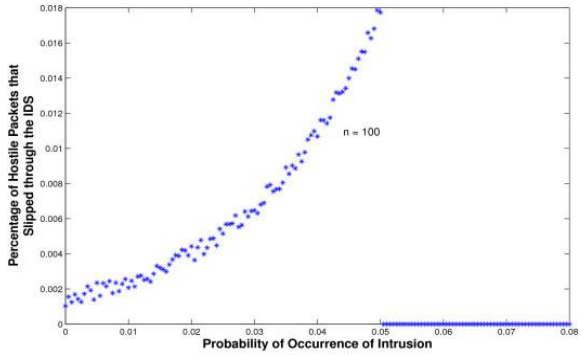



Figure 8- The percentage of hostile packets that "slipped" through the IDS as a function of PH, for n = 50.

Finally, when an outright intrusive attack is launched, the value of $P_H$ increases substantially above the threshold (which was chosen to be 0.25 for the n = 1 case and 0.05 for the n = 50 case). As the graphs in Figure 7 shows, all the malicious packets were indeed detected and quarantined as PH exceeded the calculated threshold. It is important to note here that there is essentially no difference between the data in Figures 7and 8. The value of n was irrelevant, as the graphs clearly show, since all the intrusive activity in these tests originated from an external source (i.e., no intrusive activity originated from within the LAN). It is to be added that those 9 hostile packets were non-contiguous packets. As is well known, three hostile packets for a single user cannot initiate any serious intrusive process on a network [18]. The above results clearly demonstrate that the selective inspection approach is a highly effective alternative to the common technique of blindly inspecting all Internet traffic. Obviously, after obtaining a rough estimate of the important threshold of PH from algorithm, the value of the threshold can be fine-tuned to meet a more lenient or a more restrictive IDS policy.

## IV. CONCLUSION

The analysis of network intrusion as markov chain model is the common notation that is necessary to complete inspect every packet entering into routers. The result is shown here fully support the analysis together with testing data, demonstrates that is inspects only a small number of first two packets predominantly as long as the probability of occurrences of intrusion that is determined by probability value. The SNORT rule substantial for software IDS solution from selective inspection of packets allows the IDS to high-speed links without stopping any packets. It is necessary for most of the time to eliminate the speed bottleneck problem without security. It is important to point out a process which explained in the previous session can be started from first packet to the next packet. The procedure for calculating $P_{th}$ is based on direct inspection of packet.

The relation between $P_{th}$ and th should be a proportionally relationship but not correct point. However the objectives of this work for obtaining a correct estimate of the occurrence intrusion but not for presize mathematical relationalship.The real problem of mathematical solution here is not meant to be highly presized.But it can be made correct presize with intrusion of experimental results. SNORT is a registered trademark of intrusion detection system.

## REFERENCES

[1] CH Sauer, KM Chandy, Computer Systems Performance Modeling. (Prentice Hall, Englewood Cliffs, NJ, 1981)
[2] H Kobayashi, Modelling and Analysis: an Introduction to System Performance Evaluation Methodology. (Addison Wesley, Reading, MA, 1978)
[3] FM Reza, An Introduction to Information Theory. (Dover, New York, NY, 1184)
[4] TM Cover, JA Thomas, Elements of Information Theory. (Wiley, New York, NY, 1189)
[5] Y Wang, C Wang, Modeling the effects of timing parameters on virus propagation. Proceedings of the 59 ACM Workshop on Rapid Malcode.61 (59)
[6] G Androulidakis, S Papavassiliou, Improving network anomaly detection via selective flow-based sampling. IET Commun. 2(9):189 (58). doi:5.549/iet-com:57041
[7] D Moore., et al, Inside the slammer worm. IEEE Sec Privacy. 1(4):18 (59). doi:5.159/MSECP.59.81556
[8] G Androulidakis, V Chatzigiannakis, S Papavassiliou, Network anomaly detection and classification via opportunistic sampling. IEEE Netw. 4(9):6(59)
[9] Hogwash Intrusion Detection System. http://hogwash.sourceforge.net/ (57)
[10] J Mai., et al, Impact of packet sampling on portscan detection. IEEE J Sel Areas Commun. 7(8):985 (56)
[11] N Hohn, D Veitch, Inverting sampled traffic. IEEE/ACM Trans Netw. 14(1):68 (56)

[12] P Barford, D Plonka, Characteristics of network traffic flow anomalies.Proceedings of the 1st ACM SIGCOMM Internet Measurement Wksp., San Francisco, CA. 69 (51)
[13] A Sridharan, T Ye, S Bhattacharyya, Connectionless Port Scan Detection on the Backbone. IEEE IPCCC Malware Wksp., Phoenix, Az. 1 (56)
[14] PZ Peebles, Probability, Random Variables, and Random Signal Principles. (McGraw Hill, New York, NY, 1189)
[15] IDS Wakeup: A collection of tools for testing network intrusion detection systems. http://www.hsc.fr/ressources/outils/idswakeup/index.html.en (57)
[16] A Botta, A Dainotti, A Pescape, Multi-Protocol and Multi-Platform Traffic Generation and Measurement. IEEE INFOCOM, Anchorage, Alaska. 8 http:// www.grid.unina.it/software/ITG/ (57)
[17] K Lan, A Hussain, D Dutta, Effect of malicious traffic on the network. Proceeding of Passive and Active Measurement Workshop (PAM). 1 (59)
[18] N.Kannaiya Raja., Centralized parallel form of pattern Matching Algorithm in packet inspection by efficient utilization of secondary memory in network processor., Published in IJCA(0975-8887)Volume 40-No.5,Feb2012.www.ijcaonline.org/archives/volume40/number5/4951-7194.

**N.Kannaiya Raja** received MCA degree from Alagappa University and ME degree in Computer Science and Engineering from Anna University Chennai in 2007 and he is pursing PhD degree in Manonmaniam Sundranar University from 2008 and joined assistant professor in various engineering collages in Tamil Nadu affiliated to Anna University and has eight years teaching experience his research work in deep packet inspection. He has been session conduct guest lecturer in various engineering in Tamil Nadu.Chair in major conference and workshops in computer vision on algorithm, network, mobile communication, image processing papers, pattern reorganization and bioinformatics. His current primary areas of research are packet inspection and network. He is interested to conduct guest lecturer in various engineering in Tamil Nadu.

**K.Babu** received B.E degree in Computer Science and Engineering from Anna University Chennai in 2006 and received ME degree in Computer Science and Engineering from Arulmigu Meenakshi Amman College of Engineering affiliated to Anna University Chennai. He has five years teaching experience in various engineering colleges in Tamil Nadu which are affiliated to Anna University and his research experience in Bioinformatics.

**A.Senthamaraiselvan** received ME degree in Computer Science and Engineering from Anna University, Government College of Technology, Coimbatore in 2005 and joined assistant professor in various engineering collages in Tamil Nadu affiliated to Anna University. He has five years teaching experience in various engineering colleges in Tamil Nadu which are affiliated to Anna University and his research experience in Bioinformatics.

**Dr.K.Arulanandam** received Ph.D. Doctorate degree in 2010 from Vinayaka Missions University. He has twelve years teaching experience in various engineering colleges in Tamil Nadu which are affiliated to Anna University and his research experience network, mobile communication networks, image processing papers and algorithm papers. Currently working in Ganadipathy Tulasi's Jain Engineering College Vellore.