Modeling and Performance Analysis of the Response Capacity for Alert Information in an Intrusion Detection System

Yong-Hee Jeon, Jung-Sook Jang, Jong-Soo Jang[†]

Catholic University of Daegu, Gyeongsan, Gyeongbuk, Korea
[†]Applied Security Group, Information Security Division, ETRI, Daejeon, Korea

Summary
In this paper, we propose an intrusion detection system(IDS) architecture which can detect and respond against the generation of abnormal traffic such as malicious code and Internet worms. We model the system, design and implement a simulator using OPNET Modeller, for the performance analysis on the response capacity of alert information in the proposed system. At first, we model the arrival process of alert information resulted from abnormal traffic. In order to model the situation in which alert information is intensively produced, we apply the IBP(Interrupted Bernoulli Process) which may represent well the burstiness of traffic. Then we perform the simulation in order to gain some quantitative understanding of the system for our performance parameters. Based on the results of the performance analysis, we analyze factors which may hinder in accelerating the speed of security node, and would like to present some methods to enhance performance.

Key words:
IDS, Performance Evaluation, Simulation, alert information

1. Introduction

With explosive Internet usage growth, security techniques that detect and deal with unauthorized access of personal information of others are becoming critical. The security market is growing in its size, in our country and abroad. The attacks using networks are becoming smarter too, so cases where network is accessed abnormally like by an Internet worm, normal computer operations are interrupted, or traffic of the network is increased to the point that entire network is paralyzed, are happening. So Gigabit Ethernet and large-bandwidth environment are becoming a reality, and exact detection, and even prevention techniques are being developed[1,2].

Research on system performance analysis is needed to construct gigabit-level intrusion detection system and to predict performance of the system. The performance of the system depends on the characteristics of the hardware of the chip, and on the operating system software. By performing system performance analysis on the intrusion detection node, the system's bottleneck can be examined, and from this, problems with packet management process can be found, and structural improvements can be made. Additionally, the performance of intrusion detection can be improved through the discovery of effective packet processing algorithms.

The purpose of this paper is to perform analysis and simulation on intrusion detection system that detects and deals with abnormal traffic like that of Internet worm. First, security node, Gigabit intrusion detection system which detects and deals with intrusion in high-speed network environment, is proposed and analyzed. Next, an alert arrival process is modeled regarding occurrence of abnormal traffic alert. In order to analyze burstiness effect in an alert arrival process, IBP(Interrupted Bernoulli Process) is applied to the alert arrival process when modeling, which is a discrete model of IPP(Interrupted Poisson Process)[3]. For analyzing system's performance of its response capacity, simulators using OPNET are designed and constructed, and simulation is done. Lastly, in high-speed and large-bandwidth environment like Gigabit Ethernet, elements that impede on the security node’s high speed are analyzed and a strategy is proposed to improve performance.

The remaining subjects of this paper are organized as following. Chapter 2 is on related research, and contains Intrusion Detection System, security policy, standard protocols, and how analysis is done and information is presented. In chapter 3, gigabit Intrusion Detection System - security node’s architecture is proposed and analyzed. In chapter 4, based on the analysis done on the security node, an alert information delivery mechanism is modeled and simulator is designed and implemented using OPNET. In chapter 5, based on the results of the simulation, performance is analyzed. Lastly, in chapter 6, conclusion and future work is presented.

Manuscript received December 5, 2007
Manuscript revised December 20, 2007

2. Related Research

2.1 Intrusion Detection System

Intrusion is defined as an act that inhibits integrity, confidentiality, or availability of computer resources, which causes havoc on computer system's security policy. Intrusion Detection System (IDS) is connected to most Intrusion Prevention Systems, and monitors in real-time intrusion due to abnormal use, misuse, etc, on network level or on host level. Depending on the type of intrusion detection, it is categorized as anomaly detection, misuse detection, etc. Normally, when the intrusion method is outside of known model, it is called anomaly detection, and when it is of known model, it is called misuse detection. Also, if it is installed on a host like a web service and detects intrusion only for the host, it is called host-based IDS, and if detection is for the entire portion of a network, it is called network-based IDS[2, 4-5].

2.2 Security Policy

Security policy system regulates and sets rules for important information and other resources governed and dispersed by a specific system. Security policy system is composed of security policy database, security policy server, and policy client. It uses security policy protocol to exchange information. As an example of such a policy, rule-based policy uses a qualifier which is like an indicator for expressing IP address, time, protocol, interception, login, warning, and permission for passage. It uses the qualifier to automatically execute a security policy[6-8].

2.3 Information report and analysis

Currently, there are two IDMEF messages: Alert and Heartbeat[9, 10].

2.3.1 Alert Class

In general, every time an analyzer detects an event it is set up to detect, it reports to its manager by sending an alert message. An alert message may be due to single or multiple detection events. The alert, as a countermeasure to external events, occurs asynchronously. Currently, an alert is classified into three types like the followings.

ToolAlert Class: contains additional information about malicious programs such as attack tools or Trojan horses, and when trying to identify these kinds of tools, it can be used by an analyzer.

· CorrelationAlert Class: contains additional information related to alert information's correlation. It is to group multiple alerts that were sent already.

· OverflowAlert Class: contains additional information about buffer overflow attack. It is to provide detailed information about buffer overflow attack, using an analyzer.

2.3.2 HeartBeat Class

It is used to show current status of an analyzer. Heartbeat is transmitted at given fixed intervals. Continuous reception of Heartbeat messages by the analyzer shows that it is currently in operation to the manager, and when there are no messages received, it means an analyzer or network connection has failed.

The current security system has been limited interoperability, so there are difficulties when detecting intrusion in a large network. Therefore, research is urgently required to improve information exchange techniques between Intrusion Detection Systems in a large distributed system.

3. Proposed Security Node Architecture Analysis

3.1 Node Architecture

With explosive Internet usage growth, high-speed and large-bandwidth networks like gigabit Ethernet are becoming a reality. Exact detection, even prevention, and security techniques are under research. Such security techniques that can deal with data based on high performance, are becoming ever more important[11-14]. As the speed of network goes up, more advanced intrusion detection technology is correspondingly needed. Thus, it is required to develop security appliance and engine that are level 10G or higher. According to ongoing system's structure, its performance analysis is very important and necessary.

In this chapter, the architecture of ongoing gigabit Intrusion Detection System security node, which is under development for intrusion detection and response in a high-speed environment, is simply analyzed. Figure 1 is a system block diagram of the security node structure, which offers a high-speed intrusion detection and response.

The security node is based on hardware, and by using Ethernet interface security card which detects high-speed signature on detection, it detects harmful network traffic and then subsequently blocks it. It also supports gigabit network speed. Not only does it support traffic monitoring, but it does real-time response capability. In addition, it can easily secure a whole system by using its stealth mode in a network. If it determines, after intrusion analysis, the traffic is harmful, as a response it will try to minimize harmful effects immediately.

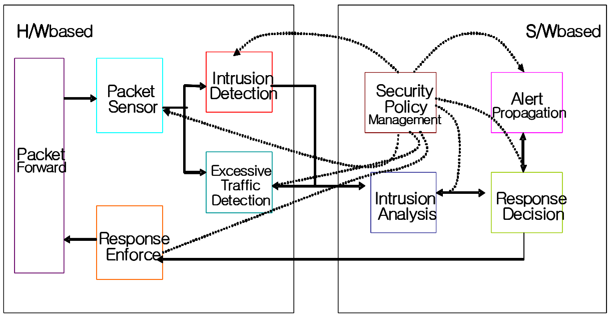


Figure 1. Proposed Security Node Block Diagram

3.2 Communication mechanism structure analysis

3.2.1 Alert message structure

Through distributed Intrusion Detection System's detection and analysis, it finds out source address and port, destination address and port, and protocol as a message with five-tuples(shown in Figure 2 below), and it delivers the information after detection[15].

Attack Type	Sour.Dest IP addr.	Sour.Dest port	Protocol	Content

Figure 2. Alert Message Transfer Format

The alert message type is as follows: attack type, source and destination address and port, protocol, and content.

3.2.2 Excessive traffic message type

The information data on excessive traffic collects BPS(Bit Per Second), PPS(Packet Per Second), and reception time of a packet by measuring packet. The information is categorized and saved according to given traffic's characteristics. It analyzes a link with a critical value to see if it is an attack like DoS(Denial of Service). For DoS attack, it is typical to attack the target host exclusively. In this case, there is a high chance that the excessive traffic is directed to edge router(situated near the attack target host)'s security node. Fluctuations of traffic fixed in a threshold value can be detected, and network can be secured by using these and network intrusion information system's alerts.

Detection of excessive traffic is divided into two stages, and governs the flow of traffic. First, using group flow management feature, it looks at traffic that might be harmful. Then, using other traffic management feature, it governs the traffic more in detail [15].

A. Aggregation Traffic Management Feature

Generally, excessive traffic occurs when single or multiple systems have a large inflow of traffic, or when one system scans multiple systems. So it is classified based on same source IP address, same destination, and destination port, and packets per second, bits per second are inspected and used for the basis for traffic analyzer's decision(Figure 3).

Source IP addr.	BPS	PPS	Ifindex

Dest. IP addr.	Dest. port	BPS	PPS	Ifindex

Figure 3. Aggregation Flow Management Message Format

B. Primitive Flow Management Feature

It separately collects and manages only suspicious traffic from group flow management feature module, and analyzes in detail the five fields, which are source IP address, destination IP address, source port number, destination port number, and protocol pattern, and manages, regarding each of the analyzed traffic, bits per second, packets per second, and final reception time(Figure 4).

Sour. IP addr.	Sour. IP	Dest. IP addr.	Dest. IP	Prot ocol	BPS	PPS	If Index	Time Stamp

Figure 4. Primitive Flow Management Message Format

Through primitive flow management feature, if it is determined that the traffic is in excess, then a message in the form below (Figure 5) is sent regarding the detected information.

SGS ID	Abno. Traf. Descript	Sour. IP addr.	Sour. port	Dest. IP addr.	Dest. port	Prot ocol	BPS	PPS	Time Stamp

Figure 5. Excessive Traffic Message Format

Excessive traffic message type is composed of security node number, technique of the abnormal traffic, source and destination addresses and ports, protocol, BPS, PPS, and packet reception time.

3.3 Security policy and response management

3.3.1 Security policy process method

The part of the information received from a security management system that pertains to policy is saved and applied to applicable modules. The security node can request security policy information from a security management system. It saves the information and applies it to applicable modules(see Figure 6)[8].

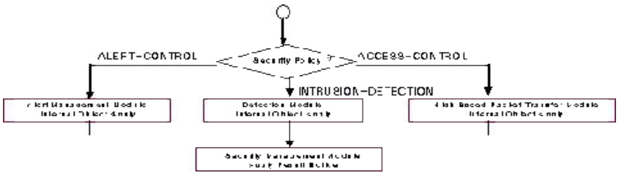

ALEFT-CONTROL
Security Policy ?
ACCESS-CONTROL
INTRUSION-DETECTION
Alert Management Module Internal Object Apply
Detection Module Internal Object Apply
High Speed Packet Transfer Module Internal Object Apply
Security Management Module Apply Result Notice

Figure 6. Proposed Security Policy Process Method

For types of security Policy, there are alert control policy, intrusion detection policy, and access control policy.

3.3.2 Intrusion Response Management

In the response selection after intrusion detection, based on the traffic measurement and various event analysis information, resulting information is sent to each of the modules. Figure 7 shows the response transmission process proposed in this paper. When the response is selected from the security management system regarding intrusion, information is sent to related modules of the security node. As for the response technique types, there are session interception, rate limiting, and event re-analysis[15].

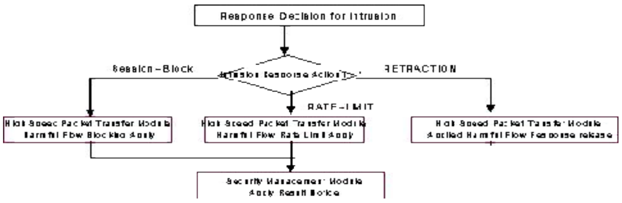

Response Decision for Intrusion
Session-Block
Intrusion Response Action?
RETRACTION
RATE-LIMIT
High Speed Packet Transfer Module Harmful Flow Blocking Apply
High Speed Packet Transfer Module Harmful Flow Rate Limit Apply
High Speed Packet Transfer Module Applied Harmful Flow Response release
Security Management Module Apply Result Notice

Figure 7. Proposed Response Decision Process Method

3.4 Alert Compression

The compression of alert for distributed intrusion detection system is to reduce a lot of alert data rate through filtering and collectivizing in order to decrease system's processing-load. The process of compression first collects the alert data, processes and saves it. Through analysis of a lot of saved data content, filtering is done by address, protocol, port number, and header information whereby alert can be deleted or allowed. Collectivizing performs compression of data by adding count information to one of the alert data for one that is created the same, and it also provides a technique for analyzing information on excessive traffic. The compression method is shown in Table 1[16,17].

Table 1. Reduction Method

Method	Format
Compression	[A, A, A, ..., A] => A
Suppression	[A, B, p(A)<p(B)] => φ
Count	[n X A] => B
Generalization	[A, A⊂B] => B
Specialization	[A, A⊃B] => B

4. Modeling and Simulator Implementation

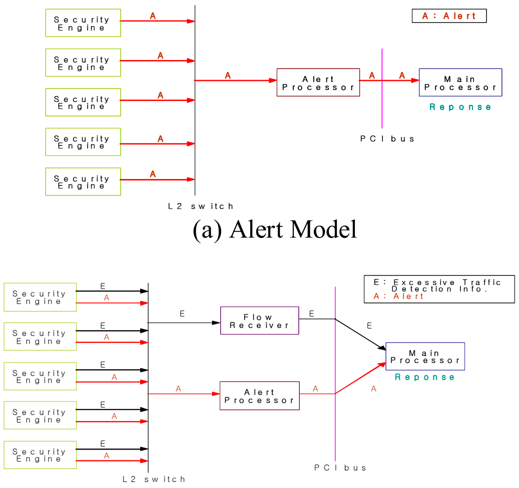
In this chapter, in order to measure performance based on the security node structure analysis, a system is modeled and simulator is implemented.

4.1 Security Node Modeling

In order to measure the performance of security node, factors that cause its performance-gigabit intrusion detection system's to lower, need to be analyzed. So in this paper, based on analysis of the security node, a system is modeled and simulator is realized to measure performance.

The security node modeling was done in two stages. Firstly, alerts that are generated by detecting all events at high-speed in hardware-based are judged by their countermeasure ability. Secondly, it is modeled in order to measure the performance in a model, which uses as an analysis for DoS attack, based on information which analyzed excessive traffic and then was summarized.

Figure 8-a is an evaluation model of main process's response ability generating alert from the security engine. Figure 8- b is a combination model of excessive traffic and alert. Because it was expected that the amount of the traffic generating excessive traffic information will be low and have a small effect on network performance, the alert model was the main model in assessing performance.



(a) Alert Model

(b) Excessive Traffic and Alert Integration Model

Figure 8. Alert Evaluation Model

4.2 Alert Arrival Process Modeling

First, for alert arrival process modeling, random process model, which is a discrete model of Poisson, is used. Random process is expressed by Bernoulli process. Also, in order to analyze the effect of burstiness, an IBP(Interrupted Bernoulli Process) model is used, which is a discrete model of IPP(Interrupted Poisson Process)[3, 18].

4.2.1 Random Process

In Bernoulli process, arrival probability in the each slot is independently p among each slot. Figure 9 shows the time slot in a random process model.

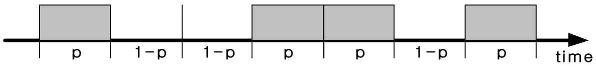


Figure 9. Time Slot at Random Process Model

4.2.2 Bursty Process

The IPP model, which the ON(active period) state that has an exponential distribution, and the OFF(silent period) state that has another independent exponential distribution appear in turn, is the main model of ON-OFF traffic. There is an IBP model which is a discrete model of IPP.

In an IBP model, the time is slotted, and it is assumed that the size is the same as a single alert packet time in the media. When a process is in the active state, either it will have probability *p* in the next slot and stay in that state, or it will have probability *1-p* and move to the pause state. If a process is in the pause state, it will have probability *q*, continue to stay in the pause state, and have *1-q* chance of moving to the active state. If a process is in the active state, each slot will have probability *a* of having alert. Figure 10 shows the time slots from a Bursty process model. In this paper, *a* is assumed to be 1.

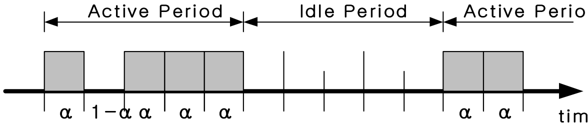


Figure 10. Time Slot at Bursty Process Model

The transition probability determinant of IBP process is shown in Equation 1.

$$\rho = \frac{1-q}{2-p-q} \tag{1}$$

If π_A and π_I are defined as each steady state probability of active or pause state, the following equation(Equation 2) can be derived from the steady state equation $\pi P = \pi$.

$$\pi_A = \frac{1-q}{2-p-q}$$
$$\pi_I = \frac{1-q}{2-p-q} \tag{2}$$

π_A is the average bandwidth or average arrival.

If d is arrival time between consecutive alerts, and d_I is the time interval taken from some pause slot to the next arrival time, squared co-efficient of variation of inter-arrival times, C^2 can be derived as in Equation 3.

$$C^2 = \frac{Var(d)}{\left[E(d^2)\right]} = \frac{(p+q)(1-p)}{(2-(p+q))^2} \tag{3}$$

In this paper, parameter C^2 is used as a measure of burstiness for alert information arrival process.

4.3 Alert Arrival Process Realization

An alert arrival process was modeled in the simulator for performance assessment using the models described above. What follows describes the realization of alert arrival process model.

4.3.1 Random Process Model

Each event arrival under a random process model has event_load parameter and is generated by Bernoulli process. Here, the event_load is a user-inputting parameter and the value range is 0<=event_load<=1. The event_load is the probability of event occurrence in optional time slot and 1-event_load is the probability which an event is not generated..

4.3.2 Bursty Process Model

The bursty process model applies IBP. The user-inputting parameters in IBP are the amount of event load and burstiness C^2's value, and a, which is the probability of including an event in every active time slot. Depending on the value a, coefficient of link utilization, the amount of event loss, event delay become different. Various parameter values can be used to analyze the special characteristics of bursty.

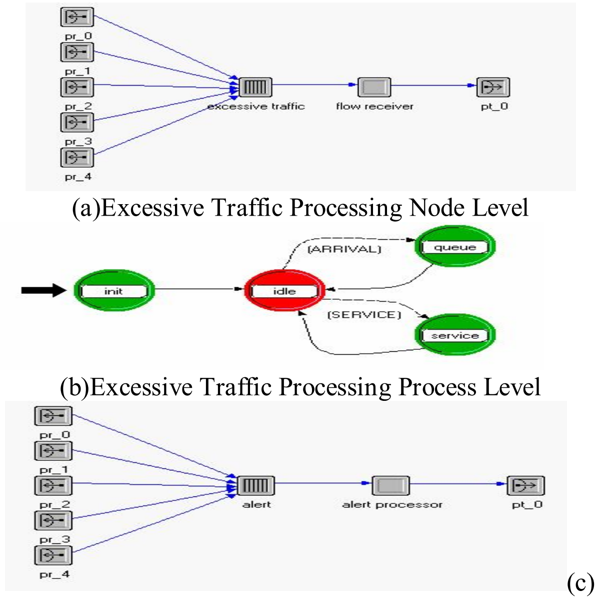
4.4 Implementation of Simulator

4.4.1 Implementation Environment

Using the modeling of security node as basis, a simulator is implemented. OPNET(Optimal NETwork), which is currently used the most in communication network simulations, is used in the development of a simulator, OPNET contains a lot of protocols that are of issue nowadays. The OPNET simulation environment and created models have been recognized by IETF.

4.4.2 Essential Parts of Simulator Implementation

Figure 11 is the result of the modeling done for purpose of assessment of alert response capacity. It constitutes the node level, which receives alert messages from five security engines and processes them, and the process level, and they both process messages regarding excess traffic.

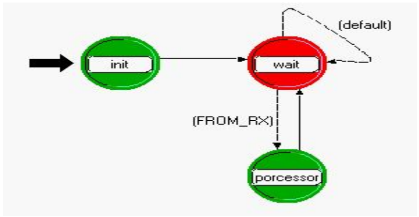


(a)Excessive Traffic Processing Node Level

(b)Excessive Traffic Processing Process Level

(c)

Alert Processing Node Level



(d) Alert Processing Process Level

Figure 11. Alert & Excessive Traffic Processing Process Modeling

4.4.3 Simulator Implementation

Figure 12 is the diagram of the complete simulator. It shows alert arrival process realization, Figure 11's alert, excessive traffic processing process, and network level which was realized using sink process.

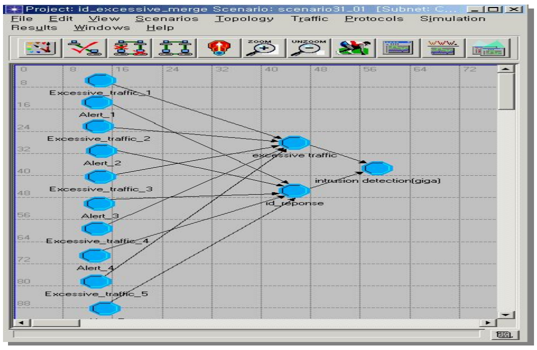

Project: id_excessive_merge Scenario: scenario31_01 [Subnet: C...
File Edit View Scenarios Topology Traffic Protocols Simulation Results Windows Help
Excessive_traffic_1
Alert_1
Excessive_traffic_2
Alert_2
Excessive_traffic_3
Alert_3
Excessive_traffic_4
Alert_4
Excessive_traffic_5
excessive traffic
intrusion detection(giga)
id_response

Figure 12. Security Node Simulator Implementation

5. Performance Analysis

5.1 Performance Analysis Scenario

With the usage of hardware-based security engine, high-speed security node generates alert in the signature-based detected data and then sends it to kernel module. The main job of the process from kernel module and application module is, after the detailed analysis of detected information, to decide, deal with, and manage a response policy. That is, the main process has to decide and deal with prompt response based on the policy, regarding alert that is generated at Giga-level speed in security engines.

The main subject of evaluation is delay performance and loss performance. The delay time is the time taken from the arrival process of alert to the time response service of main process is applied. The loss performance is calculated by counting the alerts generated from the alert arrival process, counting the arrival alerts in the sink process, and calculating the difference between them.

The parameters used to assess performance are processing speed of response, delivery rate of alert information, buffer size, etc, and are used to assess both delay and loss aspects. Table 2 shows the scenario to apply to security node's architecture.

Table 2. Performance Analysis Scenario

Scenario	Performance Analysis
1	Transfer Evaluation of Alert used at Gigabit Switch
2	Transfer Evaluation of Alert used at PCI Bus
3	Transfer Evaluation of Alert related of Buffer Size
4	Transfer Capacity Evaluation for Integration of Excessive Traffic and Alert
5	Transfer Evaluation for Alert related of Bursty Size

In the performance analysis model, alerts are generated from the five security engines, and the alert response capacity is assessed by applying the parameters. For the parameter of countermeasure ability, 100Mbps to 300Mbps were used, and the buffer size used ranged from 1,000 to 4,000.

5.2 Performance Analysis

5.2.1 Gigabit Transmission Evaluation

Figure 13 shows the evaluation of transmission performance using gigabit switch at 50% to 90% load from the alerts generated in the five security engine of assessment model.

The alerts generated using gigabit switch in the security engine represent delay performance regarding the response capacity of main processor. Depending on the number of alerts, there is a significant delay in the processing speed of response unit at 100Mbps.

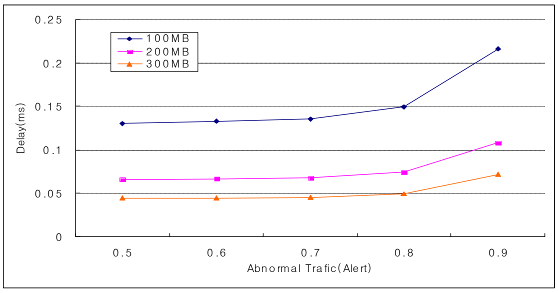


Figure 13. Alert Delay used at Gigabit Switch

5.2.2 PCI Bus Transmission Evaluation

In the case that the alert generated in security engines is transmitted to a main processor through PCI bus, a response capacity was evaluated. As a result, the result of transmission is about the same as when Gigabit switch was used. (The delay performance regarding alert information was shown to be closely related to the response processing speed, and thus we need to look into a measure to increase the response processing speed.

5.2.3 Evaluation using Different Buffer Size

Figure 14 and Figure 15 show the delay and loss of different buffer sizes with the alerts generated in the security engine at 50% to 150% load when the response unit used is running at 300Mbps and gigabit switch is transmitted. As the buffer size gets smaller, the delay is reduced; however, loss is increased. But the performance related to loss is more affected by the response processing speed than the size of buffer.

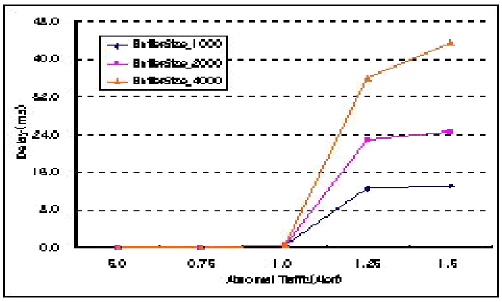


Figure 14. Alert Delay related of Buffer Size

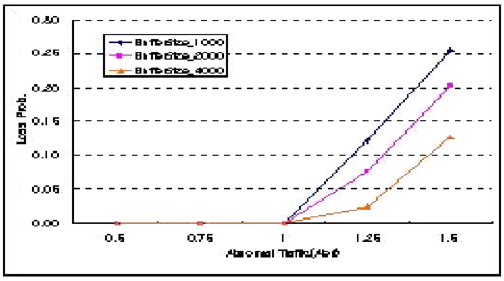


Figure 15. Alert Loss related of Buffer Size

5.2.4 Excessive Traffic and Unified Alert Transmission Performance Evaluation

Figure 16 and Figure 17 evaluated the performance, modeling excessive traffic information and alert generation in the transmission using gigabit switch regarding the whole amount of alert at 50% to 90% load generated in the five security engines. With regard to its response capacity, the performance was assessed in the combination model, which the amount of excessive traffic information increases the load by 1%, 5%, and 10%.

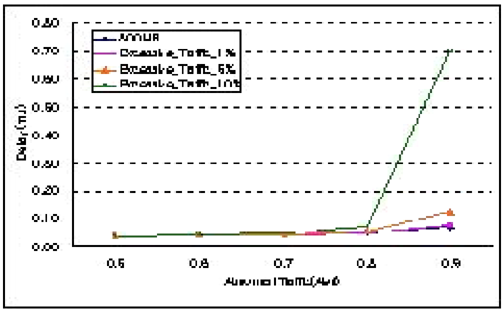


Figure 16. Excessive Traffic & Alert Transfer Delay

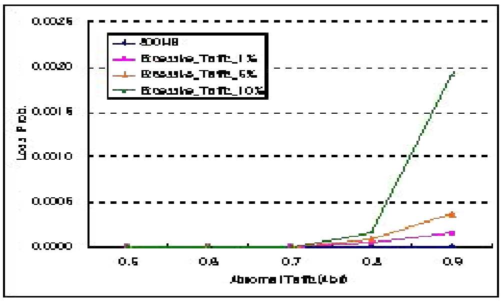


Figure 17. Excessive Traffic & Alert Transfer Loss

Figure 16 and Figure 17 show the delay and loss difference at the response capacity 300Mbps, changing the excessive traffic from detection information. As the amount of detected information increases, the delay and the loss are increased slightly.

5.2.5 Evaluation of transmission depending on burst size

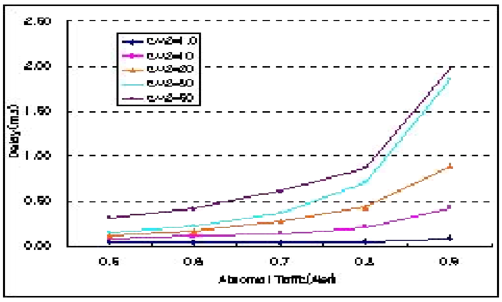


Figure 18. Alert Transfer Loss related of Bursty Size

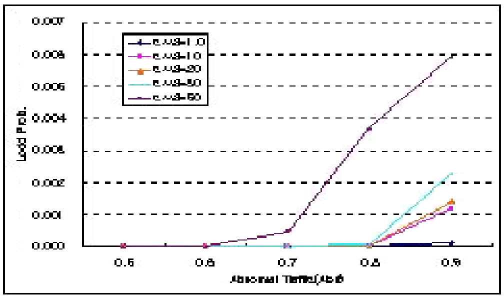


Figure 19. Alert Transfer Loss related of Bursty Size

Figure 18 and Figure 19 is to evaluate alert transmission delay and loss using Gigabit switch for the burstiness value (C^2) in the range 1 to 50, regarding the whole amount of alerts generated in the security engine of an standard model at 50% to 90% load. Figure 18 and Figure 19 show the effect on network transmission depending on the value of C^2, and Figure 18 shows that there is a significant difference of alert delay between the random process and bursty process. Figure 19 shows the loss at the buffer from excessive generation of alerts and it shows that as the size of burstiness(C^2) increases, the loss increases. Especially, as the burstiness value (C^2) gets higher and the alert load increases, the delay is increased abruptly and loss is increased.

Therefore, if distributed DoS(Denial of Service) attack is intentionally performed for a certain period, the generation of excessive alert would greatly affect the performance of security node.

6. Conclusions and Future Work

The explosive growth of Internet usage did bring many changes not only to an individual life but to the entire society, and Internet has become more important. Using networks, business documents, financial transaction, and personal information are being done and sold illegally, and associated unlawful hacking causes a serious problem in the society. To solve this kind of problems, it is becoming more important to secure at the system and network level, and especially the research of intrusion detection systems that detect unlawful intrusion, are very active.

In this paper, as hi-speed security node, the architecture of security gateway - gigabit intrusion detection system was proposed, and the system was modeled to evaluate its performance. By analyzing the structure of Gigabit security node at the security node level, the performance of communication message transmission between a hardware-based component and a software-based component seen as a bottleneck point, was evaluated. The alert message generated in the intrusion detection engine and the arrival process of excessive traffic information generated in the excessive traffic detector, were analyzed and compared to each other by using the two models, random process model and bursty process model(IBP). The OPNET was used in realizing a simulator, and the evaluation of performance were used as a parameter, using various factors that can assess the performance of communication system.

In the hardware-based hi-speed intrusion detection of the security node, intrusion detection is done at hi-speed, but as software-based processing is done, based on this detection information, the performance of overall security node depends on the response processing capacity of response devices.

The design of next-generation intrusion detection system not only proposes exact detection, but also it provides prevention and has comprehensive response techniques and high performance. It was found that the hardware-based component could be used to detect harmful traffic and generate alert at gigabit speed, but because the software-based response capacity couldn't handle at gigabit speed, the performance was degraded. For effective response, it is revealed in this paper that research on the response capacity was further necessary.

In order to enhance the performance of response capacity, techniques described in the paragraph 3.4, such as filtering and compression of security alerts, can be used as one of the methods. It is believed that research on hi-speed traffic monitoring technique and eventually cooperation methods in large-scale network environment, are necessary.

References

[1] Jai Sundar Balasubramaniyan, Jose Omar Garcia Fernandez, David Isacoff, Eugene Spafford, and Diego Zamboni, "An architecture for intrusion detection using autonomous agents". In Proceedings of the Fourteenth Annual Computer Security Applications Conference, pages 13-24, IEEE Computer Society, December 1998.

[2] Carl Endorf, Eugene Schultz, and Jim Mellander, Intrusion Detection & Prevention, McGraw-Hill, 2004.

[3] Electronics & Telecommunications Research Institute. Research Report, "Study on Traffic Generator architecture for Distributed Router Performance Analysis", December 1995.

[4] Rajeev Gopalakrishna, "A Framework for Distributed Intrusion Detection using Interest-Driven Cooperating Agents", CERIAS Tech. Report 2001-44, Purdue University, 2001.

[5] Joseph Barrus and Neil C. Rowe. A distributed autonomous-agent network-intrusion detection and response system. In Proceedings of Command and Control Research and Technology Symposium, Monterey, CA, pages 577-586, June 1998.

[6] IETF, RFC 3084, "COPS Usage for Policy Provisioning (COPS-PR)", March 2001.

[7] IETF, RFC 2251, "Lightweight Directory Access Protocol (v3)", December 1997.

[8] IETF RFC 2748, “ The COPS(Common Open Policy Service) Protocol", Jan 2000 .

[9] D. Curry, H. Debar, "Intrusion Detection Message Exchange Format Data Model and Extensible Markup Language (XML) Document Type Definition", IETF Internet Draft, draft-ietf-idwg-idmef-xml-07.txt, Jun 2002.

[10] H. Debar, D. Curry, B. Feinstein, "The Intrusion Detection Message Exchange Format", IETF Internet Draft, draft-ietf-idwg-idmef-xml-14, January 2005.

[11] Kruegel, C., Valeur, F., Vigna, G. and Kemmerer, R. "Stateful intrusion detection for high-speed networks", In Proceedings of the IEEE Symposium in Security and Privacy, pp. 266-274, 2002.

[12] ISS. RealSecure Gigabit Network Sensor. http://www.iss.net/products_serivces/enterprise_protection/rsnetwork/gigabitsensor.php, September, 2002.

[13] CISCO. CISCO Intrusion Detection System. Technical Information, November, 2001.

[14] M. Roesch. "Snort-Lightweight Intrusion Detection for Networks". In Proceedings of the USENIX LISA '99 Conference, November, 1999.

[15] Electronics & Telecommunications Research Institute, Technical Data v1.0, 2003.

[16] Frederic Cuppens, Alexander Mierge, "Alert Correlation in a Cooperative Intrusion Detection Framework", IEEE Symposium on Security and Privacy 2002.

[17] H. Debar and A. Wespi, "Aggregation and Correlation of Intrusion Detection Alerts", RAID 2001, LNCS 2212, pp. 85-103, 2001.

[18] Ichiro Ide, "Superposition of Interrupted Poisson Process and its application to packetized voice multiplexer", in ITC-12, pp. 1399-1405, Turin, 1988.

Yong-Hee Jeon received the B.S degree in Electrical Engineering from Korea University in 1978 and the M.S and Ph. D degrees in Computer Engineering from North Carolina State University at Raleigh, NC, USA, in 1989 and 1992, respectively. From 1978 to 1985, he worked at Samsung and KOPEC(Korea Power Engineering Co.). Before joining the faculty at CUD in 1994, he worked at ETRI(Electronics and Telecommunications Research Institute) from 1992 to 1994. Currently, he is a Professor at the School of Computer and Information Communications Engineering in Catholic University of Daegu(CUD), Gyeongsan, Korea.

Jung-Sook Jang received the B.S degree in Computer Engineering from Kyungil University in 1991. She received her M.S and Ph. D degrees in Computer Engineering from Catholic University of Daegu in Gyeongsan, ROK, in 1995 and 2004, respectively. From 2004, she is an IT Professor at the School of Computer and Information Communications Engineering in Catholic University of Daegu(CUD), Gyeongsan, Kore**a.**

Jong-Soo Jang received the B.S and M.S degrees in Electronics Engineering from Kyungpook National University in 1984 and 1986, respectively. He received his Ph. D degree in Computer Engineering from Chungbuk National University in 2000. Since 1989, he has received the B.S and M.S degrees in Electronics Engineering from Kyungpook National University in 1984 and 1986, respectively. He received his Ph. D degree in Computer Engineering from Chungbuk National University in 2000. Since 1989, he has been working with ETRI, Daejeon, Korea and now is the Director of Applied Security Group.