# Detecting Denial of Service using BENEF Model: An Alternative Approach

Urupoj Kanlayasiri, Surasak Sanguanpong, and Yuen Poovarawan

Applied Network Research Group
Department of Computer Engineering
Kasetsart University, Chatuchak, Bangkok, Thailand 10900
Phone (66-2) 942-8555 Ext.1433 Fax (66-2) 579-6245
E-Mail: {g4265106, nguan, yuen}@ku.ac.th

## Abstract

Computer security is a seriously concern topic for both computer system and networking. To handle various intrusive actions, an intrusion detection system can be used for detecting and countermeasuring the computer attacks. In this paper, we propose a BENEF model of network-based intrusion detection to detect Denial-of-Service (DoS) attack. This model relies on significant parameters of anomalous packets, network information, system behavior, and rule-based decision technique. The first prototype, a network-based intrusion detection system employing BENEF model, is developing and testing concurrently. The system architecture is mainly composed of three components: Feature Selector, Pre-Detector, and Decision Engine.

**Keywords**: network-based intrusion detection system, Denial-of-Service, TCP SYN flooding, rule-based detection

## 1 Introduction and Related works

In response to the growth of Denial-of-Service (DoS) attacks in computer network, it is very essential to have a tool for detecting and handling the computer intrusion [1]. An intrusion detection system or IDS is one of several ways to perform this task. IDS is an automated system intended to detect computer intrusions. The main goal of IDS is to identify, preferably in real time, unauthorized use, misuse, and abuse of computer systems by both system insiders and external penetration [2]. It also operates at network and host level for detecting various attacks. There are two domains of intrusion detecting techniques based on the detection method such as misuse detection [3] and anomaly detection [4]. Misuse or knowledge-based is an attempt to recognize the well-known flaws or vulnerabilities of software or computer system. It can detect the general attack signatures that stem from the known holes. Anomaly or behavior-based detection, on the other hand, can be identified intrusions by unusual behavior of operations.

The main advantage of misuse detection is that it makes very low false alarms. However, it also has several drawbacks [5]. First, it is a very difficult job to gather all of attack signatures. Second, to maintain and transform known

attack patterns to proper database is the time-consuming task. Finally, a variety of environments that relate to the penetrations is so complex and have a lot of details, for example, various operating systems, network configurations, hardware platforms, and program applications. Nowadays, there are many intrusion detection systems that employ misuse approach to design and develop in both research and commercial product such as expert system, signature analysis, Petri nets, and state-transition analysis.

Expert system [6] contains rules or a set of rules that describe the attack. The attack events are transformed into the semantic signification in expert system. And then, the inference engine will draw conclusion utilizing these rules and facts. Rule-based language is a rudimentary tool for describing knowledge that experts have collected about attacks. In addition, model-based approach [7] depicts the behavior of attack such as attacker's goal and the actions to accomplish the goal, etc.

Signature analysis has a little different comparing with an expert system. It transforms the known signature to the information that has the same format as the system generated. In this manner, the signature will be compared with audit trails directly. Petri Nets is the attempt to represent the attack signature into generality, conceptual simplicity, and graphical representation. The security officer or system administrator can describe the attack behavior in writing and integrate into intrusion detection system. This approach was employed by IDIOT [3]; a misuse intrusion detection system using Colored Petri Nets (CPN) developed at Purdue University. State-transition analysis is a technique proposed by [8]. This technique is conceptually identical to model-based approach but it represents the attack as state-transition diagrams.

Traditionally, the audit source location distinguishes among IDSs based on the kind of input information they analyzed. There are two categories such as host-based [9] and network-based [2] intrusion detection system. The host-based IDS monitors a single host machine employing the audit trails of a host operating system as a main source of input. It was regarded as a forerunner of the network-based intrusion detection system. The host-based IDSs, which have been widely developed in the past several years, can detect both anomaly and misuse behaviors. Generally, they often appear as the system embedded in a risky machine. The network-based IDS monitors any numbers of hosts in network segment. It peruses the audit trails of multiple hosts and network traffic to identify the intrusion signatures. This approach is a stand-alone system that can detect an intruder invaded into any systems via a computer network. Unlike the host-based IDS, it does not depend upon any operating system.

However, it is very difficult, perhaps impossible in some cases, to build an IDS that can completely detect all kinds of intrusions. Although many approaches were presented, there is no one best solution or technique for constructing the perfect system. The system may lead either "false-positive" or "false-negative" errors [10] because of uncertain decisions. False-positive error is the mistake of the system that appears when IDS classifies an action as anomalous or a possible intrusion when it is a legitimate action. A false-negative error occurs when an actual intrusive action is allowed to pass as non-intrusive behavior.

In our approach, we propose the BENEF model and formal framework of network-based intrusion detection system architecture. The key idea of this model is to scrutinize an extensive set of features that were extracted from network packets. Additionally, it also utilizes network configuration, information of network environment, and system behavior to aid the decision. To identify intrusion, the model uses threshold detection [12] and rule-based fuzzy-logic technique [11].

The rest of this paper is organized as follows. In Section 2, we describe the behavior and characteristic of DoS attack and basic concept and overall architecture of BENEF model. Section 3 and Section 4, we describe the detail of Feature Selector and Pre-Detector component respectively. In Section 5, we give the details of Decision Engine. Finally, in Section 6, we give the conclusion and future works.

# 2 Methodology

## 2.1 Denial of Service

Denial of Service (DoS) [13] is an attack class that has been used for a long time. It is a method intended to exhaust the network and station resources [14]. TCP SYN flooding is one of the common attack type and becoming a growing concern. The attacker exploits system by using a vulnerability of three-way handshake in TCP connection request operation.

Firstly, the attacker makes a lot of connection requests with spoofed address to a server. These requests are the TCP SYN packet with an unreachable source IP address. Then, the server sends back SYN/ACK message that never reach the sender. As a result, TCP connection buffers of server are allocated and rapidly exhaust. Hence, new legitimate connection can not be established. This above operation is shown in Figure 1.

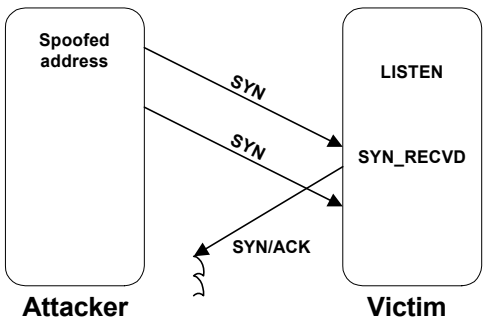


**Figure 1: TCP SYN flooding attack**

## 2.2 Overall system

The BENEF (Behavior Statistic, Network Information Base, and Fuzzy-logic Decision) model can be categorized into misuse detection. This model is intended to detect the DoS attack. In this paper, we emphasize only the TCP SYN flooding as a case study. The overall system mainly comprises three components: Feature Selector (FS), Pre-Detector (PD), and Decision Engine (DE). The system is shown in Figure2.

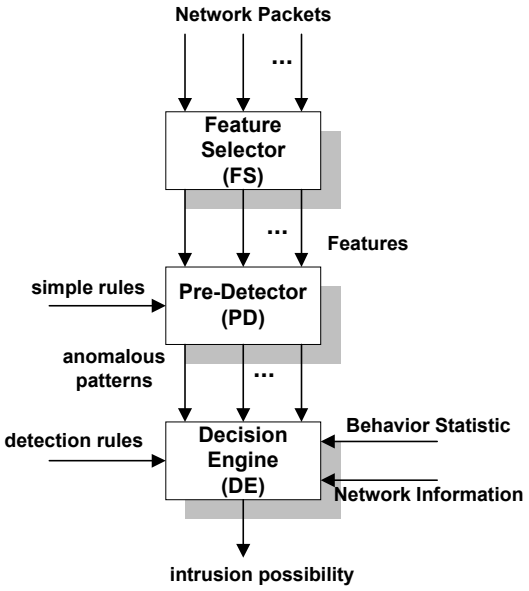


**Figure 2: Model architecture**

In our approach, a detection model for network-based intrusion detection system is proposed. The key idea is to scrutinize an extensive set of features that were extracted from network packets. Additionally, the system employs network configuration, environment information, and system behavior to aid the decision. We use threshold detection and rule-based fuzzy-logic technique to conclude final decision.

## 3 Feature Selector

The intrusion detection operation starts at FS component. It captures all packets in Ethernet local network. These packets are grouped by received station (destination host) and only SYN packets that never complete three-way handshaking are selected. Then, the system will extract the significant parameters (features) and some information from these packets. These nine significant features (K) can be defined as the system (SYN, DA, DP, SA, SP, dW, dSEQ, dT, N) as follows.

**K = (SYN, DA, DP, SA, SP, dW, dSEQ, dT, N)**

Each parameter has the following meaning:

- **SYN** is a flag on TCP header. It identifies a connection request.
- **DA** is a destination IP address of packet.
- **DP** is the destination port number.
- **SA** is a source IP address of packet.
- **SP** is the source port number.
- **dW** is an interval value of TCP window size of two consecutive packets.
- **dSEQ** is an interval value of TCP sequence numbers of two consecutive packets.
- **dT** is the interval time of two adjacent packets.
- **N** is the number of anomalous packets.

## 4 Pre-Detector

The Pre-Detector (PD) analyzes events with a set of detection rule. To detect the action of TCP SYN flooding, only five features are considered in the pre-detector phase. They are **DP**, **SA**, **SP**, **dW**, and **dSEQ**. The PD classifies individual packets based on **DP** as a primary criterion plus the other four features in various combinations as secondary criteria. Each rule is assigned an alert level; a higher of alert level indicates the more possibility of intrusion. Some portions of rules are shown as follows.

**Rule 1.**
IF (All **DP** of packets are same) AND
(All **SA** of packets are same) AND
(All **SP** of packets are same) AND
(All **dW** of packets are same) AND
(All **dSEQ** of packets are same)
THEN
Alert level 5

**Rule 2.**
IF (All **DP** of packets are same) AND
(All **SA** of packets are same) AND
(All **SP** of packets are same) AND
(All **dW** of packets are same)
THEN
Alert level 4

**Rule 3.**
IF (All **DP** of packets are same) AND
(All **dW** of packets are same) AND
(All **dSEQ** of packets are same)
THEN
Alert level 3

## 5 Decision Engine

### 5.1 Network Information

To achieve a more powerful detection, network information is required. In our case study, we maintain IP-MAC address table. IP-MAC address table contains pairs of IP and MAC addresses of all hosts in local network. For some intrusive situations such as in the case of IP spoofing [15], intrusion behavior can be clearly detected by inspecting network information.

The main proof of the following is to illustrate that some situations of IP spoofing can be detected by consulting IP-MAC address table. We now present the definitions, axioms, and lemmas that are necessary to prove that some IP address

spoofing can be detected by inspecting IP-MAC address table.

**Definition of IP address spoofing:** Any packets containing spoofed IP address of host (excluding IP gateway addresses) were sent from an attacker to a victim.

**Property 1.** A packet, which its IP address is not in the table, must have a MAC address of the gateway.

**Property 2.** A packet, which its IP address is in the table, must have its MAC address corresponding to the table.

**Property 3.** A host inside a network segment must have a unique pair of IP address and MAC address.

**Property 4.** The table maintains all pairs of IP address and MAC address of hosts inside a network segment.

**Property 5.** IP address and MAC address in table must be checked the validity and availability.

We now introduce two axioms described the case of insider and outsider attack.

**Axiom 1.** There exist only two ways in which an inside attacker spoofs its IP address:

(1) An attacker spoofs its IP address as a machine, which has IP address outside the network segment.
(2) An attacker spoofs its IP address as a machine, which has IP address inside the network segment.

**Axiom 2**[*]**.** There exist only a way in which an outside attacker spoofs its IP address as a machine, which has IP address inside the network segment.

**Lemma 1.** *For a detection technique following this model, the insider attack must be detected.*

**Proof.** From **Axiom 1**, there exist only two ways in which an inside attacker spoofs its IP address:

(1) An attacker spoofs its IP address as a machine, which has IP address outside the network segment.

    This can be easily proved by the **Property 1**. The gateway utilizes an IP forwarding mechanism by filling the source MAC address with its MAC address.

(2) An attacker spoofs its IP address as a machine, which has IP address inside the network segment

    From the **Property 2, 3, 4** and **5;** if it is found that a pair of IP address and MAC address of a packet is not corresponded with the entry in the table, we can conclude that the address was spoofed. Meanwhile, we maintain the validity and availability of the address pair all the time.

**Lemma 2.** *For a detection technique following this model, the outsider attack must be detected.*

**Proof.** From **Axiom 2**, There exist only a way in which an outside attacker spoofs its IP address as a machine, which has IP address inside the network segment.

This can be proved by the same way as **Lemma 1** using **Property 2, 3, 4**, and **5**. Because, in this case, the packet contains an IP address inside the network segment, but its MAC address is equal to the MAC address of the gateway. This address pair can not be found in the table. We,

[*] Note that, from the Axiom 2, it is very impractical under our model to detect the event; an outside attacker spoofs its IP address as a machine, which has IP address outside the network segment.

therefore, conclude that outsider attack can be detected.

## 5.2 Fuzzy rule-based decision

Most attacks normally have a noticeable signature. In our system, we interested in two features that are **dT** and **N** (in Section 3). We arrange an insider attack experiment to find out proper values for these features. A TCP SYN flooding is generated under 3 different traffic load environments: light load (no bandwidth consumption), moderated load (30% of bandwidth utilization), and heavy load (more than 65% of bandwidth utilization). We found that the victim machine is crashed when **N** is equal to 129 regardless of the bandwidth utilization. The **dT**, however, plays less important role in our experiments. The **dT** less than 20 milliseconds may crashed the victim machine.

Decision engine employs rule-based fuzzy principle to decide what pattern is an intrusion. The output is in the form of percentage of intrusion possibility. In our case study, there are two significant features, **dT** and **N**. The fuzzy rule-based system starts with the fuzzification. We assign membership values to each feature that are derived from experiment [22]. Membership function defines values into five levels, namely, very low (VL), low (L), medium (M), high (H), and very high (VH).

Additionally, the variable R is the result of detection. We create a set of rules in an IF-THEN form. These rules are derived from our experiments in detection of TCP SYN flooding attack and the surveying of many hacking reports [16, 17] and other analysis reports [15, 18, 19, 20, 21]. Some portions of rules are described as follows.

1: IF **dT**=VL AND **N**=VL THEN R=L
2: IF **dT**=VL AND **N**=L THEN R=M
3: IF **dT**=VL AND **N**=M THEN R=M

We calculate the results using MATLAB [23]. The result surface is shown in Figure 3. In this three-dimension graph, values on X-axis and Y-axis represent the feature **N** and **dT** respectively. The values on Z-axis also show an intrusion possibility.

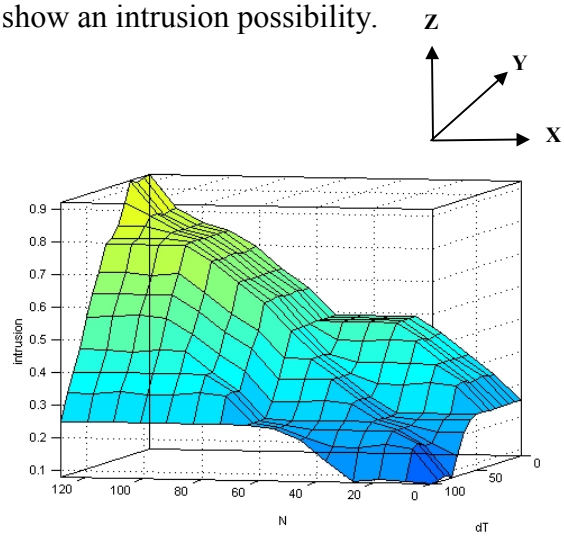


**Figure 3: Intrusion surface calculated from dT and N**

## 6 Conclusion and Future works

We propose a powerful network-based intrusion detection model for detecting TCP SYN flooding attack. The key idea of our model is to scrutinize an extensive set of features that were extracted from network packets. Additionally, the system employs network configuration, network environment information, and system behavior to aid the decision. It also uses threshold detection and rule-based fuzzy-logic technique to conclude final decision. This paper describes the progression and formal framework of our system that the prototype followed this model is developing and testing concurrently.

Our future work will focus on a development of the system cooperated with a completed operation of anomaly detection and refinement of fuzzy rule sets for a precise decision. Furthermore, the proper network information base and adequate statistics of system behavior will be adapted for better detection.

## 7 Acknowledgement

We would like to thank very much Pakorn Waewsawangwongse from the University of York, UK for providing us the good idea design of rule-based fuzzy decision technique. Thank for the helping of Chalermkon Chongsanguan and Naruchit Chinawong for contributing us some testing data as well as developing the first prototype and evaluating our approach. This work is partially funded by KURDI research grant SRU-2.43

## References

[1] Kumar, S. *Classification and Detection of Computer Intrusions*, Ph.D. Thesis, Department of Computer Sciences, Purdue University, W.Lafayette, 1995.

[2] Mukherjee, B., Heberlein, L. and Levitt, K. *Network Intrusion Detection*, IEEE Network, 8(3):26-41, May/June 1994.

[3] Kumar, S. and Spafford, E. *A pattern matching model for misuse intrusion detection*, Proceedings of the 17th National Security Conference, pp.11-21, October 1994.

[4] Spirakis, P., Katsikas, S., Gritzalis, D., Allegre, F., Darzentas, J., Gigante, C., Karagiannis, D., Kess, P., Putkonen, H., and Spyrou, T. *SECURENET: a network-oriented intelligent intrusion prevention and detection system*, Network Security Journal 1(1) 1994.

[5] Debar, H., Dacier, M., and Wespi, A. *Towards a Taxonomy of Intrusion detection Systems*, Computer Network 31:805-822, 1999.

[6] Lunt, T. and Jagannathan, R. *A prototype real-time intrusion-detection expert system*, Proceedings of the Symposium on Security and Privacy, pp.59-66, 1998.

[7] Garvey, T. and Lunt, T. *Model-based Intrusion Detection*, Proceedings of the 14th National Computer Security Conference, pp.372-385, October 1991.

[8] Porras, P. and Kemmerer, R. *Penetration state transition analysis-a rule-based intrusion detection approach*, Proceedings of the 8th Annual Computer Security Applications Conference, pp.220-229, November 1992.

[9] Kanlayasiri, U., Sanguanpong, S., and Jaratmanachot W. *A Rule-based Approach for Port* Scanning Detection, Proceedings of the 23rd Electrical Engineering Conference, Chiang Mai Thailand, pp. 485-488, November 2000.

[10] Crosbie, M. and Spafford, G. *Active Defense of a Computer System using Autonomous Agents*, Technical Report No. 95-008, Purdue University, February 1995.

[11] Kanlayasiri, U. and Sanguanpong, S. *Network-based Intrusion Detection Model for Detecting TCP SYN flooding*, Proceedings of the 4th National Computer Science and Engineering Conference, Bangkok, Thailand, pp.148-153, November 2000.

[12] Jackson, K., DuBois, D. and Stalling, C. *An Expert System Application for Network Intrusion Detection*, Proceedings of the 14th National Computer Security Conference. pp.215-225, October 1991.

[13] Lee, G. *Denial-of-Service Attacks Rip the Internet*, IEEE COMPUTER, Vol.19, No.4, pp. 12-17, April 2000.

[14] Chuba, C., Krsul, I., Khun, M., Spafford, E., Sundram, A., and Zamboni, D. *Analysis of Denial of Service Attack on TCP*, IEEE Symposium on Security and Privacy, 1997.

[15] Todd, L. and Bishop, M. *ATTACK CLASS: ADDRESS SPOOFING*, Department of Computer Science, University of California at Davis, CA.

[16] deamon9. *Project Neptune*, Phrack Magazine, 7(48), 1996.

[17] Northcutt, S. *Network Intrusion Detection An Analyst's Handbook*, NewRiders, 1999.
[18] Girardin, L. *An Eye on Network Intruder-Administrator Shootouts*, Proceedings of the Workshop on Intrusion Detection and Network Monitoring, April 1999.
[19] Green, J., Marchette, D., Northcutt, S. and Ralph, B. *Analysis Techniques for Detecting Coordinated Attacks and Probes*, Proceedings of the Workshop on Intrusion Detection and Network Monitoring, April 1999.
[20] Bellovin, S. *Security Problems in the TCP/IP Protocol Suite*, Computer Communication Review, Vol.19, No. 2, pp.32-48, April 1989.
[21] Ricciulli, L., Lincoln, P. and Kakkar, P. *TCP SYN Flooding Defense*, SRI International and University of Pennsylvania.
[22] Kanlayasiri, U., Sanguanpong, S., Chongsanguan, C., and Chinawong, N. *An Active Defense of Denial-of-Service using Cooperative System*, Proceedings of The 7th International Workshop on Academic Information Networks and Systems, Bangkok Thailand 2000.
[23] The MathWorks, Inc. *Documentation of MATLAB version 5.3.0.10183 R(11)*, January 21, 1999.