

# Using Genetic Algorithm for Network Status Learning and Worm Virus Detection Scheme<sup>\*</sup>

Donghyun Lim<sup>1</sup>, Jinwook Chung<sup>1</sup>, and Seongjin Ahn<sup>2,\*\*</sup>

<sup>1</sup> Dept. of Computer Engineering, Sungkyunkwan Univ.,  
300 ChunChun-Dong JangAn-Gu, Suwon, South Korea, 440-746  
{dhl1m, jwchung}@songgang.skku.ac.kr

<sup>2</sup> Dept. of Computer Education, Sungkyunkwan Univ.,  
53 MyungRyun-Dong JongRo-Gu, Seoul, South Korea, 110-745  
sjahn@comedu.skku.ac.kr

**Abstract.** This paper tries to propose the worm virus detection system that focuses on many connection attempts, more frequently occurring in the process of scanning than their common transmission processes. And this paper tries to determine the critical value of connection attempt by using the ordinary time network traffic learning technique which applies the genetic algorithm in order to ensure accurate detection of virus, depending on the status of network. This system can reduce the damage from worm virus more quickly than the pattern-founded worm virus detection system because it applies the common characteristics of worm viruses to detect them, and the criteria for judgment can be altered in its application though the network may change.

## 1 Introduction

Recently, many efforts are being made to detect and isolate worm virus which threaten the reliability and stability of network resources. Some of the most noticeable examples include the network resource and security management architecture which uses the network interception algorithm [6], the network access control system that applies the architect address counterfeiting and VLAN filtering [7], and network security management which uses ARP counterfeiting [8], and so on.

Worm virus is defined as a malicious code which is characterized by its active dissemination through the network with some help or without any assistance of human being [1]. For the worm virus to be spread there should be a scanning process to search for and select the object to be attacked. The host infected with the worm virus in that scanning process tends to have dramatically increased number of IP address which is communicating with the unit of time. Though several methods have been proposed to detect the worm virus, most of those methods which analyze and compare the patterns of specific worm virus have

---

<sup>\*</sup> This work was supported by grant No. R01-2004-000-10618-0 from the Basic Research Program of the Korea Science and Engineering Foundation.

<sup>\*\*</sup> Corresponding author.

the drawbacks of not being able to detect the virus if the pattern of attack changes or a new set of attack happens. In respond to that, this paper tries to detect the worm virus by applying the specific characteristics [4] of scanning that attempts to make many connections within a short time mentioned in the above.

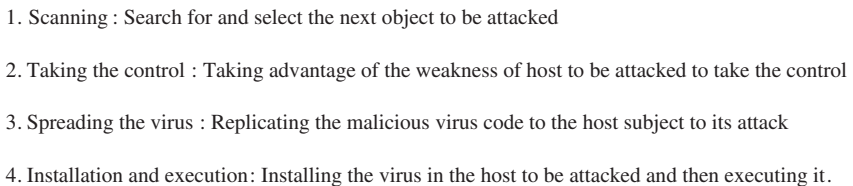
And it is necessary to study the network traffic to ensure accurate detection of virus as the number of scanning by worm virus can change depending on the status of network. For that, this researcher tries to propose the system that adapts itself to the network status by using genetic algorithm (GA).

The Section 2 of this paper examines the characteristics, detection techniques, Intrusion Detection System and genetic algorithm of worm virus as part of related study, and the Section 3 explains worm virus detection technique that uses the characteristics of scanning process. The Section 4 explains the learning technique which uses the genetic algorithm. Finally, the Section 5 draws the conclusion.

## 2 Related Study

### 2.1 Analysis on the Characteristics of Worm Virus

Worm virus is a program that takes advantage of the weakness of unspecific system from remote places to replicate itself and disseminate to other system. Generally, it referred to the program that replicates itself within the memory, but recently it refers to the program that replicates itself through the network on computer. The worm virus that replicates itself has the optimal environment condition to operate in the Internet, and that poses a great problem. Some of the most noticeable worm viruses include the Code Red, Nimda worm and so forth.

- 
1. Scanning : Search for and select the next object to be attacked
  2. Taking the control : Taking advantage of the weakness of host to be attacked to take the control
  3. Spreading the virus : Replicating the malicious virus code to the host subject to its attack
  4. Installation and execution: Installing the virus in the host to be attacked and then executing it.

**Fig. 1.** Propagation Processes of Worm Virus

The operation of worm virus is classified into direct worm and indirect worm at large, depending on the method of operation [2]. Direct worm virus takes the control on its own to spread the virus, whereas indirect worm relies on other means of transmission, such as email, to spread the virus indirectly. The direct worm usually takes advantage of the weakness of system such as buffer overflow, has a relatively broader scope of dissemination than the indirect worm, and is

very fast in spreading the virus. Meanwhile the indirect worm usually spreads the virus via the following process.

## 2.2 Intrusion Detection System

The Intrusion Detection System [5] is a type of information protection system that detects the abnormal use, abuse and misuse which compromises the confidentiality, integrity and availability of system resources, and automatically takes countermeasures or transmits the alarm message to the manager to cope with the intrusion. It is a security system that monitors the use of network or system and detects the intrusion in real time by applying the intrusion pattern database and expert system, etc, a function beyond the ordinary access control. In other words, the Intrusion Detection System is a software that detects and cope with those illegal trespassing quickly, and various software exist, ranging from simple log file analysis to complicated real time Intrusion Detection System.

The intrusion detection technique is classified into the misuse detection and anomaly detection at large. Generally, the misuse detection defines the known behavior from the attack or abnormal behavior and determines the intrusion if the collected data matches the defined behavior. Anomaly detection is also called statistical detection technique, and is a system that informs of the detected intrusion if an event which causes dramatic change or rare event happens by defining the normal and usual status as the criteria for the comparison.

In detecting the intrusion, there are problems of false positive and false negative. False positive means that what is not the intrusion is determined as the intrusion and false negative means that the actual intrusion fails to be detected. The former causes users to complain due to unnecessary policy and may undermine the productivity, while the latter fails to achieve the original goal. Therefore, Intrusion Detection System should remove the false negative and false positive as much as possible.

## 2.3 Genetic Algorithm

Genetic algorithm is the optimization technique that uses the mechanism of genetics and evolution in the nature to solve the problem. The genetic algorithm was introduced for the first time in the "Adaptation on Natural and Artificial Systems" written by John Holland in 1975.

If the information is transmitted to the living thing through genes from parents, the genetic information of individual which possesses greater adaptive power to cope with the environment than other individuals is primarily transmitted to the following generation. That is because the individual with inferior adaptive power is short-lived and cannot multiply. At the same time, species with weak adaptive power is wed out naturally. Based on that principle, individuals highly capable of adapting themselves to the environment multiply generation after generation. This is the basic principle of heredity and evolution.

Genetic algorithm expresses the possible solutions to problems as the data structure of defined pattern, and then creates better solutions by changing them gradually. In other words, it expresses possible solutions to problems as chromosomes and then changes them gradually to create better solutions. Each possible solution is deemed to be the organism or individual, and their collection is called 'population'.

One individual is composed of one or various chromosomes and the operators that change the chromosomes are called genetic operators. Basic operator includes the selection which selects the individual subject to the crossover, the crossover which replaces the genes between two chromosomes to generate new individual, and mutation which changes specific area of gene based on probability.

### 3 Worm Virus Detection Technique

There are very wide ranging types of worm viruses and their pattern of operation is very far ranging. Detection plan should be established on the basis of their operating patterns to detect various worm viruses. The character of scanning process which is among the method of worm viruses' operation is used in this paper to detect the worm virus. The process of looking for the next target is required because worm viruses can spread themselves. This process creates the IP address on a random basis to search for the IP address to be attacked next time.

At this time, the connection to IP address which is not actually used can be attempted because the IP address is created randomly. Moreover, a lot of IP addresses are searched within a short time for the fast dissemination speed of worm virus. Using this character, worm virus can be detected on the basis of the number of IP address that the host is communicating with within the unit of time. At this time, though a method to install the agent system in all hosts can be considered, I propose the method that installs the agent system at the location where all traffics of network pass in order to analyze all traffics. Agent system collects and analyzes the packet and compares the source IP address and destination IP address to investigate the number of the communicating IP address for each managing IP address.

In this paper, the packing monitoring is performed for three items as shown in Fig. 2 to ensure the maximum detection with the minimum analysis. First, analyze the address resolution protocol (ARP)[3] packet and investigate the case where the host tries to approach other host. Here, it is not the number of ARP-requesting packet but the source address and destination address of ARP-requesting packet that should be investigated. This host can judge that the scanning is being performed inside the network if same source address requests MAC value for countless destination addresses. Specifically, if the own IP address is the virus that fixes the a.b.c and creates d randomly to search for the following destination address in the event that the own IP address is a.b.c.d, this method assures an easy detection. This is an improved ARP detection method, and it was considered that the ARP requesting pack is not to be transmitted if

MAC address is already possessed. It is the method that is used to investigate the number of host that is in process of linking in case of the network with same source address and destination address. In this method, only the source of IP address and the address of destination are compared and other fields are not to be investigated in order to improve the detection function at the maximum. This final method is used in scanning through the external network, and includes both methods in the above, and considered that most of worm viruses take advantage of similar weakness. Considering a vast majority of worm viruses tend to attack the known port of Window Operating System such as 80, 135, 445 port, I used the method that analyzes only the packet for specific port among the collected packets. Compare the source and destination IP address and record the number of the connecting host for the IP address subject to the management if the packet is collected, the port is investigated and included in the port subject to the investigation.

```

1. packet ← New Captured Packet
2. if (packet = ARP)
3.   then Database .arp_mark[packet.source_ip][packet.target_ip] = checked
4. else if (packet = IP)
5.   then if (packet.source_ip ∈ Managed IP List and packet.target_ip ∈ Managed IP List )
6.     then Database .ip[packet.source_ip][packet.target_ip] = checked
7.   if (packet = TCP) or (packet = UDP)
8.     then if (packet.target_port = WeakPort List )
9.       then if (packet.target_ip ∉ Database .port[packet.source_ip] List )
10.        add packet.target_ip in Database .port[packet.source_ip] List

```

**Fig. 2.** Algorithm of Worm Virus Detection (Packet Monitoring)

After it collect information through packet monitoring as Fig. 2, it set up critical value numer about ARP, IP and a connection host of port and compare with collection data about each IP address in a managed object IP address list as Fig. 3. Perceive to a host infected with a Worm virus than the value which set up if large.

## 4 Learning Technique That Applies the Genetic Algorithm (GA)

The detection of worm virus relies on the APP, IP, number of the connection host of port, and critical value. If the critical value of those numbers are too low, false positive occurs which mixes up even the normal connection with the worm,

```

1.  $\alpha \leftarrow$  ARP threshold value,  $\beta \leftarrow$  IP threshold value,  $\gamma \leftarrow$  Port threshold value
2. for ip in Managed IP addresses List
3.   arp_count  $\leftarrow$  count checked database.ARP[ip]
4.   if (arp_count >  $\alpha$ )
5.     then notify ip is infected.
6.   else
7.     initialize database.ARP[ip]
8.   ip_count  $\leftarrow$  count checked database.IP[ip]
9.   if (ip_count >  $\beta$ )
10.    then notify ip is infected.
11.  else
12.    initialize database.IP[ip]
13.  port_count  $\leftarrow$  count checked database.PORT[ip]
14.  if (port_count >  $\gamma$ )
15.    then notify ip is infected.
16.  else
17.    initialize database.PORT[ip]

```

**Fig. 3.** Algorithm of Worm Virus Detection (Counting Connections)

whereas false negative occurs which fails to detect the worm if the critical value is too high. GA is used to determine the optimal value of those critical values. In the proposed system, GA learns the input value of fuzzy controller. Learning data is required for the learning. The learning data is composed by the pair of [the number of connection host of ARP/IP/Port] [Whether there is any worm virus infection] The learning program learns the fuzzy value by using learning data shown in Table 1. In order to learn the fuzzy input value, the fuzzy section should be converted to a genetic type than can be used by GA as Fig. 4.

Table 1. Structure of Learning Data

IP Count	Infection or Not	ARP Count	Infection or Not	Port Count	Infection or Not
84	True	12	False	37	False
12	False	77	True	61	True
1	False	50	True	48	False
8	False	23	False	72	True
..	..	..	..	..	..

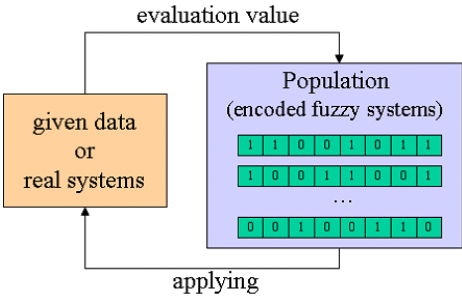


Fig. 4. Diagram Showing the Structure of Fuzzy System That Uses GA

The proposed system had the fuzzy section in the form of trapezoid, and used each point as the genetic value. Each section in Fig. 5 can be expressed by the following equation.

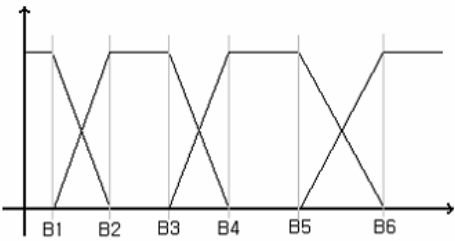


Fig. 5. Section of Genetic Type

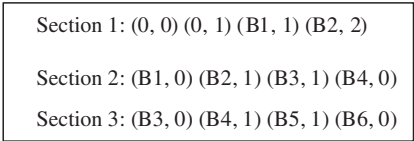


Fig. 6. Indication of Genetic Type of Trapezoid Section

The section is divided into 9 sections all told, and a total of 16 points from B1 to B16 are created which define the section in this case. Therefore, the gene

RULE 0: IF IP\_Count is B1 THEN Low;

RULE 1: IF IP\_Count is B2 THEN Low;

RULE 2: IF IP\_Count is B3 THEN Low;

RULE 3: IF IP\_Count is B4 THEN Slightly\_High;

RULE 4: IF IP\_Count is B5 THEN Slightly\_High;

RULE 5: IF IP\_Count is B6 THEN High;

RULE 6: IF IP\_Count is B7 THEN High;

RULE 7: IF IP\_Count is B8 THEN Very\_High;

RULE 8: IF IP\_Count is B9 THEN Very\_High;

**Fig. 7.** Proposed Fuzzy Rule

If the result from the input learning data is False:

- If the result value of fuzzy controller is Low, the fitness is +5 scores.
- If the result value of fuzzy controller is Slightly\_High, the fitness is +2 scores.
- If the result value of fuzzy controller is High, the fitness is +1 score.
- If the result value of fuzzy controller is Very\_High, the fitness is 0 score.

If the result from the input learning data is True:

- If the result value of fuzzy controller is Low, the fitness is 0 score.
- If the result value of fuzzy controller is Slightly\_High, the fitness is +1 score.
- If the result value of fuzzy controller is High, the fitness is +3 score.
- If the result value of fuzzy controller is Very\_High, the fitness is +5 scores.

**Fig. 8.** Fitness evaluation of Fuzzy input

can be defined in the form of [B1 B2 B3 ... B15 B16]. In creating the initial individual, 16 integers are created randomly, and then they become the genetic types if they are arranged in ascending order. Rule is not separately learned but fixed because the input variable is only one. The proposed fuzzy rule is like this;

The result value is divided into 4 values which indicate the possibility of worm infection: Low, Slightly\_High, High, Very\_High. As for the learning, the fitness is determined depending on the result obtained by fuzzy controller after the genetic



type is converted to input value, and crossover for the superior individual is performed. The fitness of fuzzy input value is evaluated by the following method.

The evaluation algorithm of GA derives the fitness in the wake of the learning of all learning data. If the fitness is obtained, the crossover of superior individual is performed. The individual is based on the RouletteWheel which has the possibility of selection proportional to the fitness, and the selection value is defined as 4. The crossover method is 2 point crossover, and the mutation rate was defined to be 0.15

If the aforesaid fitness method was applied, definitive evaluation (if false, low, or if true, very high) would be made to get the top score in case that GA is sure that the input value is certainly the worm virus. If GA is unsure, it will make ambiguous evaluation (slightly\_High) to avoid the cut in scores as much as possible. As a result, Slightly\_High will be learned at the boundary line where the infection of worm virus is determined, and if that value is exceeded, High or Very\_High will be determined as to the fuzzy section.

## 5 Conclusion

This paper made it possible to detect fast detection of worm virus by using the function that detects the host infected with worm virus. In addition, this paper proposed the method that enables the optimized system itself to determine the worm virus status by using the fuzzy value for critical value, instead of the absolute value that users enter, and learning through genetic algorithm.

This researcher tries to experiment to find how much efficiency the system that learns the critical value on its own through the precise simulation on the network identical to real situation, which applies this system, can achieve in the future, compared to the existing structure which requires users to input. Moreover, it seems that it will identify and complement the problem arising from the simulation, and through that process, more improved system may be developed.

## References

1. Darrell M. Kienzle and Matthew C. Elder, "Recent worms: a survey and trends," Proceedings of the 2003 ACM workshop on Rapid Malcode, 2003
2. Jason C. Hung, Kuan-Cheng Lin, Anthony Y. Chang, Nigel H. Lin and Louis H. Lin, "A bahavior-based anti-worm system," In Proceedings on AINA'03, China, 2003
3. David C. Plummer, "An ethernet address resolution protocol," RFC 826, 1982
4. Vincent Berk and George Bakos, "Designing a framework for active worm detection on global networks," in Proceedings of the First IEEE International Workshop on Information Assurance, 2003
5. Wagner D. and Dean R., "Intrusion detection via static analysis," in Proceedings of 2001 IEEE Symposium on Security and Privacy, 2001

6. Jahwan Koo, Seongjin Ahn, Jinwook Chung, "Network blocking algorithm and architecture for network resource and security management," in Proceedings of International Scientific-Practical Conference "Problems of Operation of Information Networks", 2004
7. Wonwoo Choi, Hyuncheol Kim, Seongjin Ahn and Jinwook Chung, "A network access control system using on address spoofing and VLAN filtering," The 4th Asia Pacific International Symposium on Information Technology, 2005
8. Kyohyeok Kwon, Seongjin Ahn and Jinwook Chung, "Network security management using ARP spoofing," in Proceedings of ICCSA 2004, 2004
9. D. E. Goldberg, *Genetic Algorithm in Search, Optimization, and Machine Learning*, Addison-Wesley publishing company, Inc. 1989
10. D. Dasgupta and F. A. Gonzalez, "An intelligent decision support system for intrusion detection and response," In Proceedings of International Workshop on Mathematical Methods, Models and Architecture for Computer Networks Security, pp 1-14, May 2001
11. M. Crosbie and G. Spafford, "Applying genetic programmings of to intrusion detection," In Proceedings of AAAI Symposium on Genetic Programming, pp. 1-8 November 1995.