International Journal of Engineering & Technology, 8 (1.4) (2019) 257-265

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Comprehensive Threat Response Model for the Prevention of Website Intrusion

Yong-joon Lee[1], Jeong-min Lee[2], Jin-seob Kim[3], Jae-pyo Park[4]*

[1]Defense Security Institute, 89-10, Seosomun-ro, Jung-gu, Seoul 03742, Republic of Korea
[2]Korea Information & Security Agency, Venture Tower, 135 Jungdae-ro, Songpa-gu, Seoul 05717, Republic of Korea
[3]Information Security Division, Shinhanbank, 102-903, Seoulsilipdaero 75, Dongdaemun-gu, Seoul, Republic of Korea
[4]Department of Information Security, Soongsil University, Seoul 07027, Republic of Korea
**Corresponding author E-mail: pjerry@ssu.ac.kr*

Abstract

The purpose of this study is to verify the effectiveness of a model capable of comprehensive defense against malicious code attacking websites by using technology to cope with website Intrusion. In this study, we suggest a comprehensive response system model against website intrusions with five security functions, for a comprehensive action against the intrusions occurring through websites. The proposed comprehensive response model against website intrusion attacks is designed to make an overall action against malicious codes through the websites using five response techniques, in terms of the following three stages: prevention, detection, and recovery. The proposed five response techniques include web vulnerability pre-checking, the detection of homepage-spread malicious codes, the detection of zombie PC through malicious code address change, the cyber treatment service, and DDoS (Distributed Denial of Service attack) cyber shelter. An empirical experiment was used for the cases of website intrusion attacks to prove that it was possible to make an efficient action. Experimental results show that 12,191 vulnerabilities have been detected through preliminary checks of web vulnerabilities and 2,932,979 zombie PCs have been blocked. We also detected 9,778,683 PCs infected with malicious code and verified the effectiveness of this proposed model. We expect it will be possible to make effective responses against the intrusion attacks using the comprehensive response system model for websites suggested in this study.

Keywords: *Website Intrusion, Website security, cyber attack, threat response, Model*

1. Introduction

There are growing cases of cyber attacks and e-financial frauds due to the spread of malicious codes through the homepages with a large number of visitors. This study suggests a comprehensive response model against website intrusion attacks to analyze the malicious codes spread through homepages and minimize the damage caused by the codes. The administrator should safely manage the website using web vulnerability pre-inspection to take action against malicious attacks on the website. The website vulnerability inspection refers to the ex-ante inspection of the vulnerabilities against hacking attacks, which could be used to make preliminary action before the website intrusions.

Hackers spread the malicious codes through the website using the website vulnerabilities, so it is necessary to make continuous monitoring of such intrusion. When a malicious code is detected, the code analysis is made and the extraction of C&C (Command & Control) information is made, where C&C makes orders to the malicious codes. When the hacker attempts to connect to C&C to change the malicious domain address, it is possible to block the command control site and the zombie PC that is infected with the malicious codes spread through the homepage, to prevent it from receiving command orders from the hacker, by making a bypass to the sinkhole server instead of C&C.

In addition, when the DDoS (Distributed Denial of Service attack) attack occurs from a PC infected with malicious codes, a cyber shelter can be operated to prevent DDoS attacks and detect the information on the PC infected with the malicious codes. The infected PC detected by the DNS sinkhole and cyber shelter provides a pop-up window for the zombie PC users to give them an alert that their PC is infected with the malicious code, and provide them with the cyber treatment service with customized vaccines. It is expected that an effective intrusion-response will be available through the comprehensive response model against the web intrusion attacks.

2. The Risk of Website by DDos Attack

A DDoS attack is a malicious cyber attack that causes overload in a short period of time by remotely controlling dozens and even millions of PCs at the same time on a particular Web site. Due to the nature of web applications of processing user-inserted information and data, unspecified individuals can easily make use of them, and they are always exposed to external attacks. Therefore, there is a high risk of being under cyber attacks. The traditional cases of intrusion are targeting the vulnerabilities of operating systems or applications, while

the recent attacks are targeting the vulnerabilities of web applications due to DDos Attack. The vulnerabilities of web applications are easier to attack from the hackers with relatively lower skills when compared to other hacking methods, and they can quickly deliver malicious codes to the users[1].
In terms of the types of hacking cases, web-related vulnerabilities have been ranked high in the list recently, such as in the cases of web-shell, file upload, SQL injection, and web application vulnerabilities, and this type of web-related intrusions are on the rise[2]. Furthermore, hacking attacks targeting Internet-based shopping malls and content-providing companies are persistently occurring. Most of such hacking attacks are using the vulnerabilities of websites operated by the companies, and they are trying to make additional attacks, such as backdoor installation, authorization acquisition, internal system vulnerability scanning, and private information intrusion, by falsifying the company websites[3].

3. Website Intrusion Response Techniques

3.1. Pre-Inspection of Web Vulnerabilities

The architecture of web vulnerability inspection is shown in Figure 1. The administrator makes regular updates of the web vulnerability inspection pattern (Vulnerabilities Information DB), and the system automatically makes vulnerability inspections for the websites registered in the web vulnerability inspection target list (Target Domain Information DB)[4-6]. The inspection module collects the information through web crawling module and saves the history, such as the inspection result (application data DB) log.
The saved inspection result has an architectural structure that it creates the inspection report through the web vulnerability inspection pattern result (Application Vulnerabilities DB), and automatically sends it to the website administrator[7,8]. It is possible to prevent the malicious code intrusion through hacking attacks by making a web vulnerability pre-inspection that provides web vulnerability inspection result and related methods to remove the malicious code[9,10].

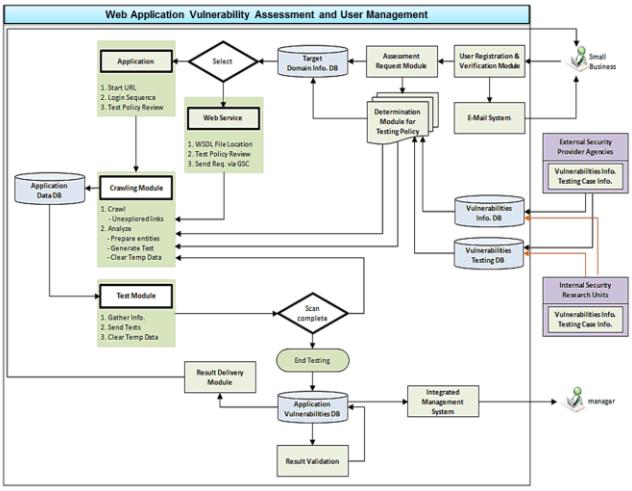

Fig. 1: Architecture of web vulnerability inspection

3.2. Detection of Website Intrusion

Figure 2 shows the architecture of homepage-spread malicious code detection. The operator makes continuous updates of the domestic domain information on the 2.5 million target homepages (Target Domain DB) and registers the information. The website inspection module (Malware Site Detection Module) makes inspections in terms of two methods, which are signature-based detection and action-based detection, and makes an overall report of the detection result.
In the signature-based detection, it inspects the main homepage information collected through the crawling method (Web Crawler Collection) regarding about 20 thousand patterns registered in the malicious code spread pattern (Malware Exploit Signature DB)[11]. In the behavior-based detection, it makes additional connections of about 500 thousand homepages with large numbers of users to the virtualized PC environment, and detects any malicious behaviors, for example, if the system has downloaded malicious codes or if the PC system has changed
As the two types of inspections are done, the history is comprehensively managed in the malicious code waypoint detection information (Malware Landing Site DB), and the security manager can request a deletion order to the homepage administrator. In addition, the automatic malicious code collection through the behavior-based inspection and security manager can connect to the homepage and collect malicious codes manually using an analysis tool [12]. The collected malicious code allows command control server (C&C) analysis through in-detail code analysis, and allows additional infected PC information by applying C&C address change (DNS sinkhole)[13].

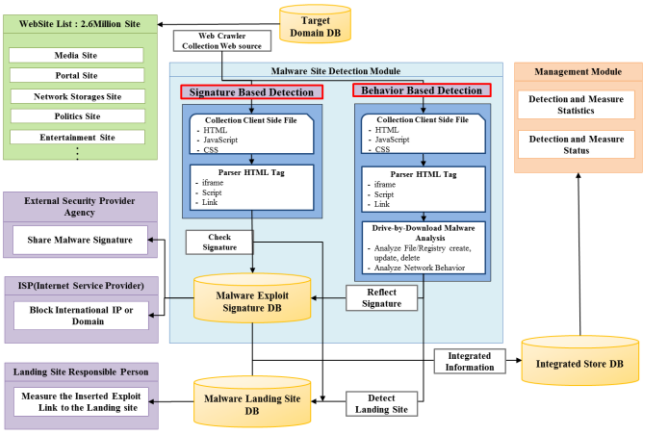

Fig. 2: Malicious Code Detection Architecture

3.3. DNS Sinkhole

The DNS sinkhole architecture for zombie PC detection is shown in Figure 3. The security manager first make regular updates of the command control server (C&C) information (C&C Domain DB) collected through the homepage-spread malicious code analysis. The updated list of command control is automatically reflected through the DNS server application module (Apply C&C domain in the DNS) of the related institutions. As the process is done, the infected PC attempts to connect to the command control site, but it is induced to the DNS sinkhole server, and the infected PC detection is available through the infected PC detection module (Zombie IP Classification Module).

The reason is as follows: when the command control server (C&C) domain IP is changed, the infected PC attempts to access the command control server (C&C) that is directed by the malicious code, but it is induced to the DNS sinkhole server for detection. The detected infected PC information is classified by the statistics module, and the infected PC's IP information is stored in the response system (Cyber Curing System) to inform the users about the infection [14].

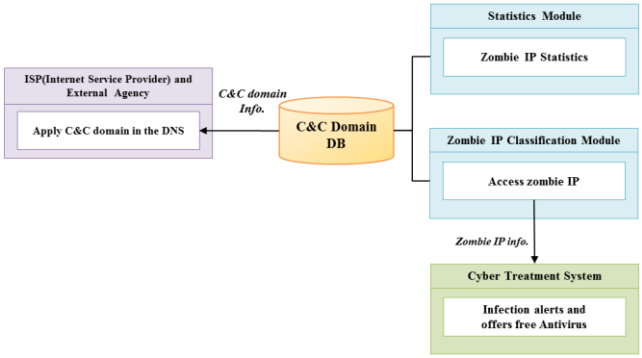

Fig. 3: Detection Architecture of Zombie PC through Malicious Domain Address Modification

3.4. DDoS Cyber Shelter

Figure 4 shows the architecture of cyber shelter. Even though the infected PC is treated with the cyber treatment service, a DDoS attack may occur onto the website by the remaining untreated infected PC. In this case, it is possible to make a primary blockade by referring to the DDoS attack pattern detection using the defense module against DDoS attacks (Anti-DDoS Defense Module), if the website attacked by the remaining zombie PC is immediately switched into a cyber shelter.

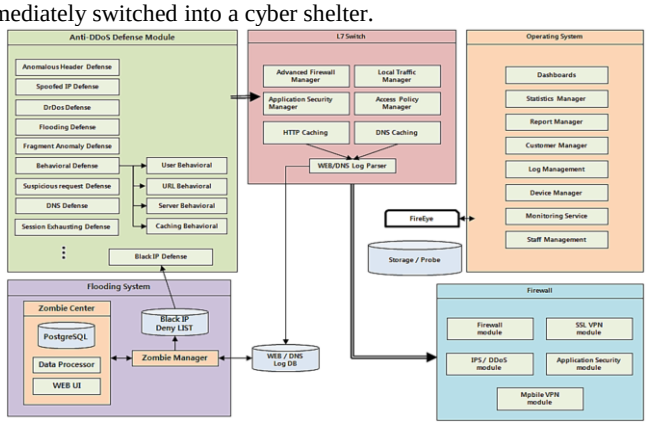

Fig. 4: DDoS Cyber Shelter Architecture

Then, the secondary defense against abnormal packets can be performed regarding the web access traffic to the cache server (L7 Switch) where the damaged website is copied. All the DDoS traffics are collected into the system for pattern analysis (Flooding System), and the list of infected PCs used for the DDoS attack is stored in the infected PC IP list (Black IP Deny List). The access line to cache the dam-

aged website information is protected by a firewall. The security manager operates anti-DDoS response, and the attacker traffic collection and analysis are controlled by the operating system[15].

3.5. Cyber Curing System

The architecture of the cyber curing system is shown in Figure 5, which provides cyber treatment services for the infected PCs that have been detected through the malicious domain address manipulation. Among the detected infected PC information (Malware Infection IP Information), the infection time information, the IPs of the infected PCs, and the malicious code information is integrated into the treatment system (Cyber Curing System DB). First of all, the malicious code samples are analyzed to develop customized vaccines (Vaccine Development). As the vaccine is created, the infected PCs' IP information and the customized vaccine are transmitted to the main ISPs[16]. The ISP then opens a pop-up window on the user PCs to help them treat the malicious codes, and the statistics and records of infected PC treatment results are then managed by the comprehensive management module (Management Module). The cyber curing system allows the prevention of recurrence and further spread of the malicious codes by making a finalized treatment of the infected PC[17].

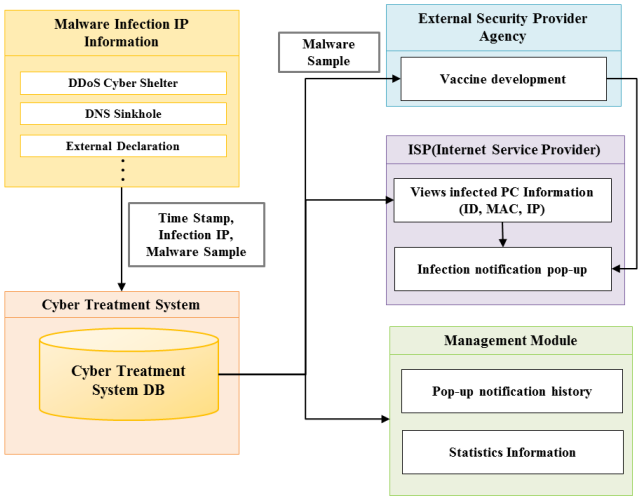

Fig. 5: Architecture of Cyber Curing Service

4. Proposed Comprehensive Response Model against Website Intrusion

In the cases of major nationwide cyber attacks, including the 3.4 DDoS attack (2011) and 3.20 cyber attack, the malicious codes were widely distributed through the homepages and network storages with a large number of visitors. As shown in Figure 6, this study proposes a comprehensive response system model against website intrusion attacks by integrating technical measures to detect malicious codes and prevent the spread of damage. The comprehensive response system model against website intrusion attacks consists of the following three stages: prevention, detection, and recovery. Also, it aims for a comprehensive action against malicious codes on the website with the following five response techniques: web vulnerability pre-checking, the detection of homepage-spread malicious codes, the detection of zombie PC through malicious code address change, the cyber treatment service, and the DDoS cyber shelter.

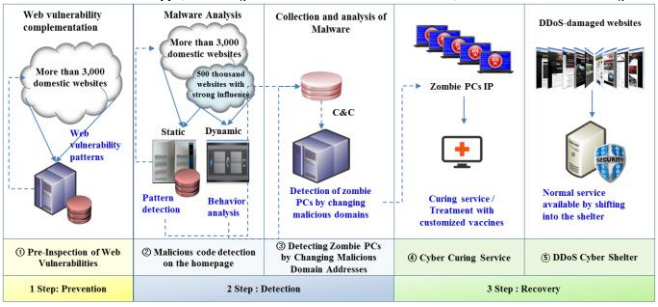

Fig. 6: Comprehensive Response System Model against Website Intrusion

The reason for using the 3-step and 5-step countermeasures in this proposal model is to provide a comprehensive response for website Intrusion treatment and prevention, which means the life cycle of the comprehensive security system of the website.

5. Experiment of the Comprehensive Response Model against Website Intrusion

5.1. Experimental Environment

In this study, a 4-years of experiment (2011 - 2014) was conducted regarding the websites operated in Korea, as shown in Table 1.

Table 1: Experimental Environment of the Comprehensive Response System Model against Website Intrusion

No.	Analysis technique	Contents
1	Pre-Inspection of Web Vulnerabilities	• Make pre-inspections for the websites in terms of web vulnerabilities to prevent hackings from the outside

		• Ex-ante prevention is possible by conducting web vulnerability inspections
2	Malicious code detection on the homepage	• Malicious code spread detection through regular inspections on the websites for domestic users • Immediate announcement to the administrator when a malicious code-spreading homepage is detected, so the administrator can delete the malicious code
3	Detecting Zombie PCs by Changing Malicious Domain Addresses	• Analysis of command control (C&C) server by analyzing the malicious codes spread through homepages, network storages, etc. • Detection and prevention of zombie IPs by using the DNS sinkhole server for the infected PCs connecting to the designated command control site
4	Cyber Curing Service	• Infection announcement for the users by collecting and analyzing the access address of the infected PC through the homepage or network storage • Provide the victims of the malicious codes with customized vaccines to allow treatment
5	DDoS Cyber Shelter	• Apply the cyber shelter when an attack occurs from the malicious code-infected PC despite all the countermeasures • Recovery of normal services of the homepage attacked by DDoS attacks by applying cyber shelter

5.2. Experimental Results

5.2.1. Web Vulnerability Pre-Inspection

This experiment was conducted as part of the government project of Korea Internet & Security Agency (KISA), which is a web site security management organization in Korea. We conducted a total of 12,191 web sites from 2011 to 2014 (4 years). Experiments were conducted after obtaining prior consent from companies that own or operate.
In this experiment, a 4-years of web vulnerability inspection was conducted. Table 2 shows the number and total of web vulnerability inspection cases by year.

Table 2: Web Vulnerability Inspection Status (2011 - 2014)

Year	2011	2012	2013	2014	Total
Number of injection	3,029	3,040	3,070	3,052	12,191

As a result of providing web vulnerability inspection service focused on the web hosting service providers and web agencies, a number of hosting operators used the web vulnerability inspection service. A total of 3,052 cases of vulnerability inspection was made in 2014, and 77,412 vulnerabilities were found and relevant complementation was recommended. With the web vulnerability inspection engine used in this study, some of the inspection criteria were removed, when they are less important, not necessary, or there were chances of causing system disability. Therefore, a quick and safe inspection could be made without causing any issues on the homepages in operation. Also, the inspection criteria were renewed each year to allow the detection of potential new types of web vulnerabilities.
As shown in Figure 7, the homepages with the request regarding web vulnerabilities were classified into the industrial types. The top five industries were internet-based services, followed by web portals, online shopping malls, software businesses, and manufacturing services.

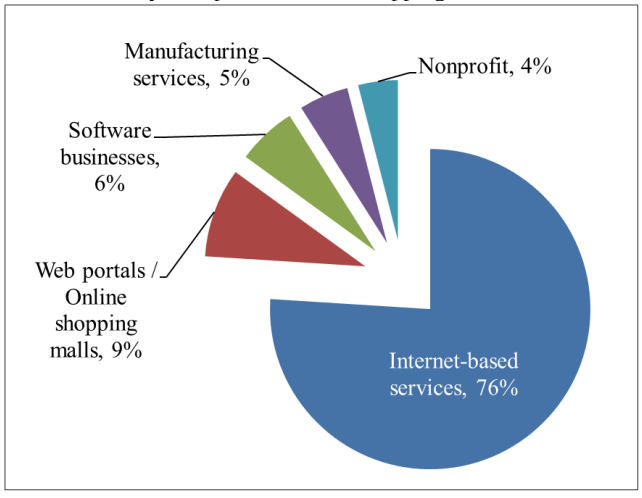

Fig. 7: Industrial Distribution of Web Vulnerability Inspection (2014)

Figure 8 shows the statistical information of the top five industries with the highest number of web vulnerabilities. Each vulnerability factor was classified into high, intermediate, and low level of significance.
The vulnerability inspection showed 41% of high-level significance, followed by 34% of intermediate-level and 25% of low-level significance. The web vulnerabilities were often caused by minor errors occurred in the process of development or management. The cross-site scripting or SQL injection issues, which were classified into high-level significance group, were not difficult to make complementation. Thus, it was suggested that the significance of the vulnerability and the difficulty of the action were not always proportional.
The vulnerability inspection report included complementary measures and references to the detected vulnerabilities. A list of technical documents was also provided in the reporting emails, and the consultation or technical support of the inspection results was provided through telephone or e-mail. However, due to the nature of the web vulnerability, most of the programs needed to be modified, and when the client company lacked developmental personnel or did not have enough time to understand the security measures, the action could not be performed or a longer time was necessary to conduct the measures.

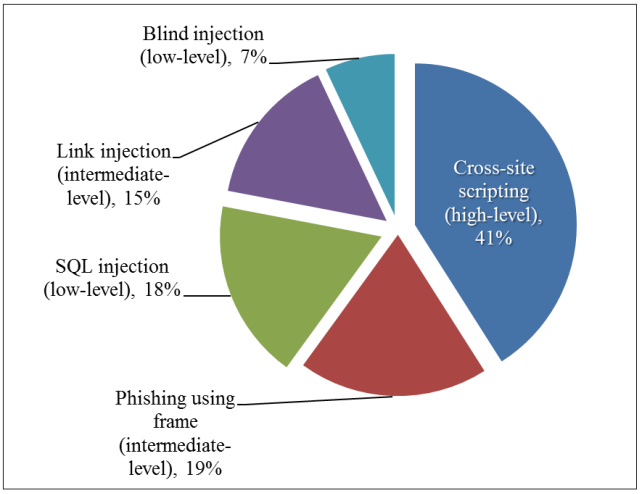

Fig. 8: Top Five Web Vulnerability Inspection Criteria Distribution (2014)

5.2.2. Website Intrusion Detection

Table 3 shows the status of homepage-spread malicious code detection for four years (2011 - 2014). The domestic homepage inspection was extended (2.3 million cases → 2.5 million cases), and the number of malicious-code hidden website detection was increased by 168.7% (17,750 cases → 47,703 cases) when compared to 2013.

Table 3: Homepage-Spread Malicious Code Detection and Countermeasures (2011 - 2014)

Year	2011	2012	2013	2014	Total
Number of Waypoint*	10,372	9,748	13,278	45,120	78,518
Number of Spreading site**	1,433	3,270	4,472	2,583	11,758
Total	11,805	13,018	17,750	47,703	90,276

* Waypoint: The homepage that indirectly distributes malicious codes by automatically linking the homepage visitors to the spreading site
** Spreading site: The homepage that directly distributes malicious codes to the homepage users

For the domestic homepages with malicious code detected, a removal request was delivered by phone or through an official document. For the foreign homepages, blocking or other relevant measures were performed through the major ISPs to prevent the spread of damage. As a result, the number of direct-spreading sites decreased by 42.2% in 2014 (4,472 cases → 2,583 cases). However, the number of waypoints that relayed the access to the spreading sites increased by 239.8% (13,278 cases → 45,120 cases) over the previous year. It was suggested that the cause of increased waypoints was that the malicious codes were directly delivered to the endpoint users, and it was difficult to conduct measures in the endpoint PCs.
Table 4 shows the information of malicious code-hidden websites in 2014. The homepages of general businesses (small and medium-sized businesses) accounted for the highest proportion of 58%, followed by others (including private websites) nonprofit websites, networks, laboratories, and college websites.

Table 4: A Classification of Homepage-Spread Malicious Codes by the Types of Institution (2011 - 2014)

Year	2011	2012	2013	2014	Total
Businesses	8,740	7,723	10,242	27,544	54,249
College	39	166	120	60	385
Nonprofit	727	820	723	4,851	7,121
Laboratories	0	23	28	1,768	1,819
Networks	647	627	779	3,079	5,132
Others	1,652	3,659	5,5858	10,401	71,570
Total	11,805	13,018	17,750	47,703	90,276

Figure 9 shows the vulnerability status of malicious code-hidden sites. It was found that CVE-2011-3544 (Java applet vulnerability), CVE-2012-4681 (Java applet vulnerability), and CVE-2013-0422 (Java applet vulnerability) were mostly abused in 2014 for the malicious activities. Also, new types of vulnerabilities (Adobe Flash Player, MS Explorer/OLE) were compositively abused to spread the malicious codes.

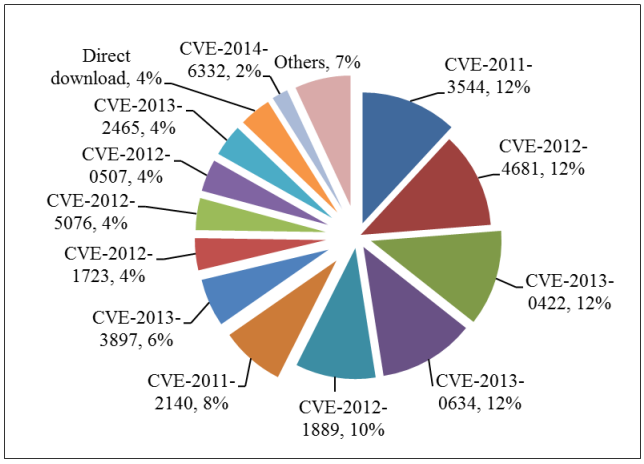

Fig. 9: Distribution of Homepage-Spread Malicious Code Vulnerability in 2014

As shown in Figure 10, in terms of the types of malicious codes, the codes for financial information leakage (38%) accounted for the highest proportion, followed by financial company pharming (17%), remote control (16%), dropper (13%), and information leakage (6%).

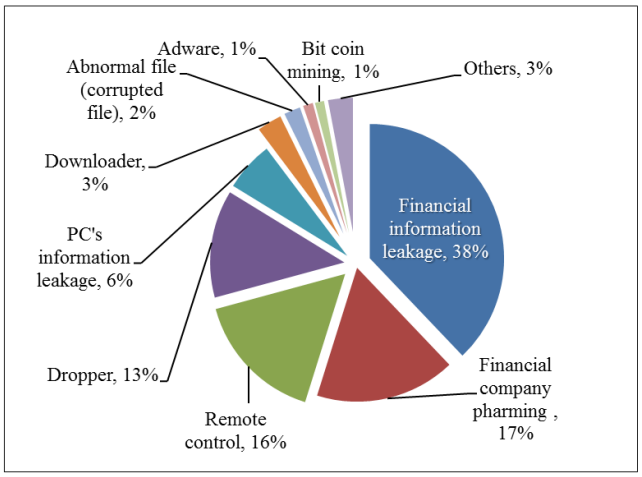

Fig. 10: Distribution of Malicious Codes by Types in 2014

The detected foreign spreading sites were blocked with assistance from the major domestic ISPs (Internet service providers) to prevent the access from domestic users. Also, for the waypoints, notifications were delivered to the homepage administrators by phone, email, or through official documents that their websites were used for the spreading of malicious codes. They were informed to remove the malicious codes and conduct proper security measures.

The five most commonly detected vulnerabilities among websites with vulnerabilities are shown in Figure 11: The cross-site scripting (XSS) accounted for 41%, phishing through frame 20%, SQL injection 15%, link injection 13%, blind SQL injection 7%, and temporary file download and others took 4%.

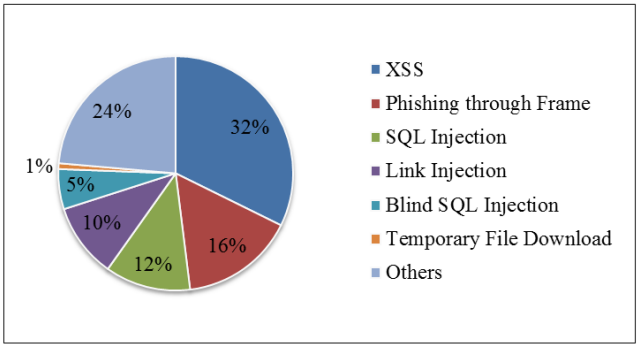

Fig. 11: Major Web Vulnerabilities of Websites

5.2.3. DNS Sinkhole

The malicious code-infected zombie PCs and C&C could be blocked by using the system that prevents receiving commands from the hacker by inducing the infected PC to the DNS sinkhole server instead of C&C server, in cooperation with major domestic ISPs. Blocking the link between the infected PC and the C&C enabled the prevention of hacker's malicious activities, such as DDoS attacks or spamming using the zombie PCs. Table 5 shows the result of zombie PC command site blocking using the DNS sinkhole.

Table 5: Zombie PC Blocking Using the Method of Malicious Domain Address Manipulation

Year	2011	2012	2013	2014	Total
Number of C&C Collection Cases	3,711	4,122	6,068	67	13,968
Number of Zombie PC Blocking Cases	1,996,822	666,107	252,969	17,081	2,932,979

5.2.4. DDoS Cyber Shelter

Since its opening, a total of 949 institutions has used the cyber shelter service until the year 2014. Among the cases, a total of 424 cases of DDoS attack defense service were provided. Also, there was recurrence prevention effect by making further measures, such as treatment and blockade, after getting the IP information of zombie PCs and attack command servers collected through the defensive process. In particular, it could detect more than 128 million zombie PCs in 2014, which was about a 250 percent increase from 486,000 cases in 2013.

Table 6: DDoS Cyber Shelter Service Status

Year	2011	2012	2013	2014	Total
Number of Service User	101	175	260	413	949
Number of DDoS Defense Cases	60	138	116	110	424
Number of IPs Used to Attack Shelter	97,444	162,508	486,811	1,285,669	2,032,432

5.2.5. Cyber Curing System

As shown in Table 7, a total of 248,281 infected PCs was notified of the infection in 2014. Also, a total of 67 customized vaccines was developed and distributed. In particular, the number of pharming malwares that aimed to leak financial information was rapidly increased in 2014. Hence, we developed and distributed customized vaccines to the zombie PCs infected with the detected pharming malicious codes.

Table 7: Cyber Curing Service Status

Year	2011	2012	2013	2014	Total
Number of Infection Notification	9,404,759	61,437	64,206	248,281	9,778,683
Number of Customized Vaccine Development	54	5	57	67	229

6. Discussion

Based on the experimental results above, we could conclude a four-years of overall analysis on the comprehensive response system against website intrusion attacks in terms of the following five characteristics.

First, a total of 12,191 web vulnerability pre-inspection measures were taken over the four years (2011 - 2014), and about 3,000 cases were voluntarily inspected each year. The checks were focused primarily on Internet-based service providers. The vulnerabilities, such as cross-site scripting, frame phishing, and SQL injection, continued throughout the experimental period.

Second, a total of 90,276 cases of website intrusion attacks was detected during the four years, and most of the malicious codes were aimed for financial fraud. Such spread of malicious codes has continued to increase rapidly, leading to the increased need for a countermeasure in this field.

Third, in terms of the zombie PCs directed to DNS sinkhole, a total of 13,968 C&Cs were blocked that led to the blocking of 2,932,979 zombie PCs. The findings suggest that a quick detection and blockade of C&C could be effective in preventing the spread of zombie PCs.

Fourth, a total of 949 small and medium-sized businesses have used the DDoS cyber shelter for four years, where 424 DDoS attacks were defended. It was about 100 cases of DDoS defense every year. The DDoS attacks were characterized by an increase in the number of attacks rather than frequency during the experimental period.

Fifth, the cyber curing system provided 9,778,683 times of notifications to the malicious code-infected PCs during the four years of the period. It was also found that the number of PCs infected by the pharming malicious codes aiming for the leakage of financial account information was on a continuous increase.

7. Conclusion

This study proposed a comprehensive response model against website intrusion attacks using the five security functions for an overall response against the intrusion occurring through websites. The comprehensive model enables a comprehensive response against the malicious codes spread through websites, using the five techniques, including the web vulnerability pre-inspection, homepage-spread malicious code detection, the detection of zombie PCs by malicious domain manipulation, cyber curing service, and DDoS cyber shelter, through the three stages of prevention, detection, and recovery.

An experiment was conducted using the proposed model for four years (2011 - 2014) to verify the effectiveness of the model, and it was proven that the proposed comprehensive response model could make effective measures against the website intrusion attacks. . Experimental results show that 12,191 vulnerabilities have been detected through preliminary checks of web vulnerabilities and 2,932,979 zombie PCs have been blocked. We also detected 9,778,683 PCs infected with malicious code and verified the effectiveness of this proposed model. We expect the comprehensive response system model, as well as the experimental and analytical results suggested in this study will be helpful in making effective responses against potential intrusion attacks in the future.

References

[1] Tiago Vieira, Carlos Serrão, "Web Applications Security and Vulnerability Analysis Financial Web Applications Security Audit – A Case Study," *International Journal of Innovative Business Strategies(IJIBS)*, 3(2), pp.86-94, 2016.

[2] Kwang-Hyoung Lee, Jae-Pyo Park, "A Software Vulnerability Analysis System using Learning for Source Code Weakness History," *Journal of The Korea Academia Industrial cooperation Society*, 18(11), pp.46-52, 2017.

[3] Provos, N., McNamee, D., Mavrommatis ,P., Wang,K. & Modadugu, N. "The ghost in the browser analysis of web-based malware," *In Proceedings of the first conference on First Workshop on Hot Topics in Understanding Botnets*, April, 2007.

[4] Provos N. , McNamee D., Mavrommatis P., Wang K., & Modadugu N. "The ghost in the browser analysis of web-based malware," *Proceedings of the first conference on First Workshop on Hot Topics in Understanding Botnets*, pp.4-4, April 2007.

[5] Kwang-Hyoung Lee and Jae-Pyo Park, "A Software Vulnerability Analysis System using Learning for Source Code Weakness History," *Journal of The Korea Academia Industrial cooperation Society*, 18(11), pp.46-52, 2017.

[6] So-Yeon Min, Chan-Suk Jung, Kwang-Hyong Lee, Eun-Sook Cho, Tae-Bok Yoon and Seung-Ho You, "Design of Comprehensive Security Vulnerability Analysis System through Efficient Inspection Method according to Necessity of Upgrading System Vulnerability," *Journal of The Korea Academia Industrial cooperation Society*, 18(7), pp.1-8. 2017.

[7] S. Wagner, D. M. Fernandez, S. Islam, and K. Lochmann, "A Security Requirements Approach for Web Systems", *Proceedings of the Quality Assessment in Web (QAW2009)*, CEUR, 2009.

[8] Sang-Hwan Oh and Tae-Eun Kim, Hwan-Kuk Kim, "Technology Analysis on Automatic Detection and Defense of SW Vulnerabilities," *Journal of The Korea Academia Industrial cooperation Society*, 18(11), pp.94-103, 2017.

[9] Jae-Hyun Lee, "Study on the OWASP and WASC-oriented Web Application Security," *Journal of Advanced Navigation Technology*, 15(3), pp.376, Jun. 2011.

[10] Jae-Chan Moon, "Vulnerability Analysis and Threat Mitigation for Secure Web Application Development," *Journal of the Korea Society of Computer and Information*, 17(2), pp.133, Feb. 2012

[11] The Open Web Application Security Project (OWASP), Available Online at http://www.owasp.org. Accessed in Sep. 2011.

[12] Joonseon Ahn, Byeongmo Chang and Eunyoung Lee,, "Quantitative Scoring System on the Importance of Software Vulnerabilities," *Journal of the Korea Institute of Information Security and Cryptology*, 25(4), pp.921-932, 2015.

[13] Boo Joong Kang, Kyoung Soo Han, Eul Gyu Im, "Malware Current Status and Detection Technology," *Journal of Communications of the Korea Information Science Society*, 30(1), pp.44-53, 2012.

[14] Korea Information & Security Agency(KISA), A method for analyzing malicious code distribution patterns, Research Report, Dec. 2010.

[15] Korea Information & Security Agency(KISA), October trends, Cyber Security Issue, Dec. 2013.

[16] Korea Information & Security Agency(KISA), Large-scale Malware distribution trend analysis report, May 2014.

[17] Korea Information & Security Agency(KISA), Internet Infringement Incident trends and responses in 2014, Feb . 2015.