DOI: 10.2298/CSIS120908026C

Two-Step Hierarchical Scheme for Detecting Detoured Attacks to the Web Server

Byungha Choi[1] and Kyungsan Cho[2]

[1] Graduate School, Dankook University
Yongin, Gyeonggi, Korea
notanything@hanmail.net
[2] Corresponding Author
Dept. of Software Science, Dankook University
Yongin, Gyeonggi, Korea
kscho@dankook.ac.kr

Abstract. In this paper, we propose an improved detection scheme to protect a Web server from detoured attacks, which disclose confidential/private information or disseminate malware codes through outbound traffic. Our scheme has a two-step hierarchy, whose detection methods are complementary to each other. The first step is a signature-based detector that uses Snort and detects the marks of disseminating malware, XSS, URL Spoofing and information leakage from the Web server. The second step is an anomaly-based detector which detects attacks by using the probability evaluation in HMM, driven by both payload and traffic characteristics of outbound packets. Through the verification analysis under the attacked Web server environment, we show that our proposed scheme improves the False Positive rate and detection efficiency for detecting detoured attacks to a Web server.

Keywords: detection scheme, two-step detection, detoured attack, signature-based, anomaly-based, outbound traffic

1. Introduction

Attacks to information systems have evolved steadily over a long time, and more Web-based attacks have replaced traditional attacks. Nowadays, more systems are reliant upon the Web server to get and exchange information through the Internet and the security shifts from lower layers of network protocol to the application layer. Thus, Web-based attacks focused on applications have become one of the most serious topics in the security field. That is, Web-based attacks focus on an application itself and functions on layer 7 of the OSI[13].

Application vulnerabilities could provide the means for malicious end users to break a system's protection mechanisms in order to take advantage of or gain access to private information or system resources. The most common Web-based attack types are SQL Injection, XSS (Cross-Site Scripting), buffer overflow, password cracking, spoofing, repudiation, information disclosure, denial of service, and evaluation of privileges[12,13].

Web-based attacks expose the weakness and vulnerability of the victim system, and disseminate malware to other hosts communicating with the victim system. Layered security systems with firewall, IDS (Intrusion Detection System) and WAF (Web Application Firewall) are provided to cope with the above attacks, and they protect the victim system very well. Traditionally, they detect external outsider threats by inspecting the traffic towards the system. Even though outsider attacks are constantly evolving and increasing, they are well detected and protected with the corresponding technical improvement. However, because an insider can directly access the Web server, these attacks should be coped with in other ways. It is found that insiders show unusual activities or abnormal behaviors when they access system resources for attacking purposes[21]. Thus, most works are based on identifying abnormal insider's behaviors and finding any significant change in insider's activities.

In addition, there are detoured attacks which bypass the traditional Web-based intrusion path. For example, there are e-mails with attached malware to insiders, or methods that use malicious USB memory or PDA with the aid of insiders and backdoor attacks. This unusual type of detoured attacks (including insider attacks) has become a more serious and common threat, and has already overtaken Web-based viruses and worm attacks[4]. Many detoured attacks to Web servers disclose confidential/private information, or disseminate malware codes to outside of the victim system, and these harms could be detected by inspecting the outbound traffic.

Thus, instead of inspecting inbound traffic, or analyzing a user's profiling and activity, we propose a scheme to inspect and detect abnormal outbound traffic caused by detoured attacks. Our proposed scheme is a two-step detection system composed of a signature-based IDS that uses Snort and an anomaly-based IDS that uses probability evaluation in HMM (Hidden Markov Model).

This paper is a revised and extended version of our previous work which was submitted to the MIST-2012 workshop[6]. The followings are major improvements to the earlier version:

1 To extend the scope of our detection, we improve the anomaly-based detection method, by adding a scheme detecting abnormal traffic characteristics from non-HTTP packets. In addition, we extend rules in the signature-based detection. With this extension, we verify the detection efficiency to backdoor attacks which produce abnormal non-HTTP traffic.
2 We evaluate our proposed two-step detection scheme with new datasets in terms of the FP (False Positive) rate, detection efficiency and detection rate. Through verification, we show that two detection methods in our proposal are complementary to each other, and our proposal is very efficient in detecting abnormal HTTP packets.
3 Most figures and tables are revised, and the evaluation results are extended, according to the above revision.

The rest of the paper is organized as follows. In Section 2, we review related work on security vulnerabilities of the Web Server and solutions to them. In Section 3, we propose our two-step hierarchical detection scheme. In Section 4, we

verify our scheme with real datasets collected under the attacked environment. In Section 5, a summary is provided.

2. Related Works

In 2010, OWASP announced the updated Top 10 most critical Web application security risks to educate about the consequences of the most important Web application security weaknesses and to provide basic techniques to protect against these high risk problem areas[17]. The WASC threat classification v2.0 shows proper classification of threats into attacks and weaknesses for a static/core view[1]. Both show the seriousness of Web-based attacks, with focus on the application layer of the protocol suite. Application vulnerabilities could provide the means for malicious end users to breach a system's protection mechanisms, in order to gain access to confidential and private information or system resources[13].

To detect Web-based outsider attacks, a layered Web security system is commonly used with firewall, IDS and Web application firewall. The security system protects the Web server from external attacks by inspecting the inbound traffic as shown in Figure 1.

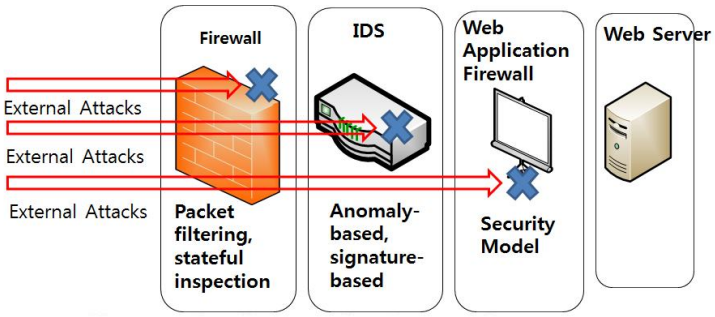

Fig. 1. Traditional Layered Security System for Web server

In the first stage of a layered security system, the firewall, which filters the specific network traffic between the network and the Web server, uses packet filtering and stateful inspection to detect simple intrusions. However, firewalls cannot prevent previously unknown attack types or insider attacks[14]. The second stage, IDS, uses signature-based or anomaly-based techniques, to protect against attacks that pass the firewall in the first stage. Signature-based IDS finds known patterns of misbehaviors in the message context, and anomaly-based IDS finds any deviation from the normal context patterns. IDS also can be classified according to the location and purpose as NIDS (Network-based

IDS) or HIDS (Host-based IDS). Mostly, NIDS is used in the second stage of a layered security system[23]. A Web application firewall filters packets by applying a set of rules. It uses a Positive Security Model or Negative Security Model or both. It filters packets which have already passed both the firewall and IDS[9].

Most current IDSs use only one of the two detection methods, signature-based detection or anomaly-based detection. However, each has its own limitations. Signature-based IDSs cannot detect any unknown attacks whereas anomaly-based IDSs cannot detect any untrained attacks. Thus, the detection accuracy of signature-based detection is extremely high, but the FP rate of anomaly-based IDSs is not negligible. Snort is a widely used IDS which allows pattern search for signature-based detection, and some works on using Snort rules in IDS have been proposed [11]. Security tools incorporating anomaly-based detection are proposed, and HMM, a statistical model of a system as a Markovian process with hidden states, has been shown to provide high level performance for detecting intrusions[5,15]. Lately, hybrid IDSs have been proposed that combine the two approaches into one system. For example, a hybrid IDS is obtained by combining packet header anomaly detection (PHAD) and network traffic anomaly detection (NETAD), or a hybrid intrusion detection system (HIDS) is configured with three sub-modules of misused detection module, anomaly detection module, and signature generation module[2,10]. However, by inspecting inbound traffic, they detect only external attacks.

Even though external attacks are constantly evolving and increasing, they are well detected and protected with the corresponding technical improvement. In fact, a layered security system pays little attention to what is happening inside the system. But, insiders show unusual activities or behaviors when they access system resources for attack purpose. Thus, information about a user's pattern of behavior and activity could be inspected for detection purposes. Most works on inside attacks are based on identifying abnormal insider's behavior, and finding any significant change in an insider's activity. Maloof and Stephens developed ELICIT, which detects insiders by inspecting insider's violating "need to know"[16]. However, this may not be enough to make a conclusion of a malicious act merely from knowing only a user's activity, and need further verification[21].

In addition to the insider attacks, detoured attacks which bypass the traditional security path to the Web server, are possible. For example, e-mail attacks with attached malware, attacks using USB memory/PDA attack with the aid of the insider, and backdoor attacks are detoured attacks. From hence, we address detour attacks as including insider attacks. If an e-mail containing a malware is sent to the insider and the insider accepts it, it causes the victim system to be remotely controlled by the outsider. If an outsider connects USB memory or PDA infected malware to a Web server with the aid of any insider, it causes detoured attack. A backdoor, which often refers to a backdoor program, is a hidden method for obtaining remote access to a computer that bypasses the traditional security mechanism. Thus, backdoor attack is a kind of detoured attack to be detected in our work. Backdoors can easily be installed on a victim

system that they aim to its exploit, thus detecting them requires considerable policies. A basic principle for backdoor detection is to find distinctive features of the activity of interest[22]. Table 1 shows possible detour attacks, and Figure 2 shows the possible detoured path of attacks described in Table 1.

Table 1. Types of Detoured Attacks

Type	Description
Through E-mail	Outsider sends e-mails containing malware procedures to the insider
Intentional Error	Insider intentionally makes programming errors
Through USB/ PDA	Insider connects contaminated USB/PDA with malware
Getting Information	Outsider gets account information from the insider
Backdoor Attacks	A hidden method of obtaining remote access to a computer system that bypasses the traditional security mechanism

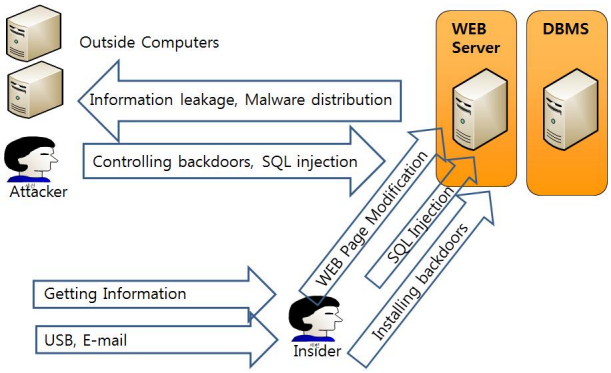

Fig. 2. Intrusion Path of Detoured Attacks

Detoured attacks to Web servers could use weaknesses of OWASP Top 10 or malware codes, thus causing altered HTML documents with Tags and JavaScript codes, SQL injection in DB, as well as altered traffic patterns. When a falsified Web page is activated by malware code, it could disclose confidential/ private information, and distribute malware codes[27]. However, conventional security solutions monitor network communication without paying much attention to outgoing traffic, due to the high processing cost of packet level network traffic analysis[24]. Lately, several works on inspecting outbound traffic for secu-

rity reasons have been proposed. For example, information leaks through HTTP were measured and detection of outbound malware traffic was proposed[3].

It is already known that most detoured attacks to a Web server show similar patterns in the HTTP outbound traffic[7], and a potential HTTP-based application-level attack exploits the features of Hypertext Markup Language (HTML)[26]. Thus, currently unknown detoured attacks could also be detected by inspecting outbound traffic. If any deviation from the normal context pattern for the specific traffic is found in the outbound traffic, it could be detected as ”attacked”. However, the outbound packets generated by detoured attacks, such as backdoor attacks, could be both HTTP packets and non-HTTP packets. It is shown that many samples use HTTP and continue with non HTTP-based damage functionality[20]. Snort also has TCP rules that have a port different from the HTTP port for non-HTTP traffic[18].

3. Proposed Detection Scheme with Two-Step Hierarchy

Our proposed scheme has a two-step hierarchy. The first step is signature-based detection using Snort and the second step is anomaly-based detection using HMM. We cannot find any other hybrid system that combines these two detection approaches to detect detoured attacks.

3.1. Overview of the proposal

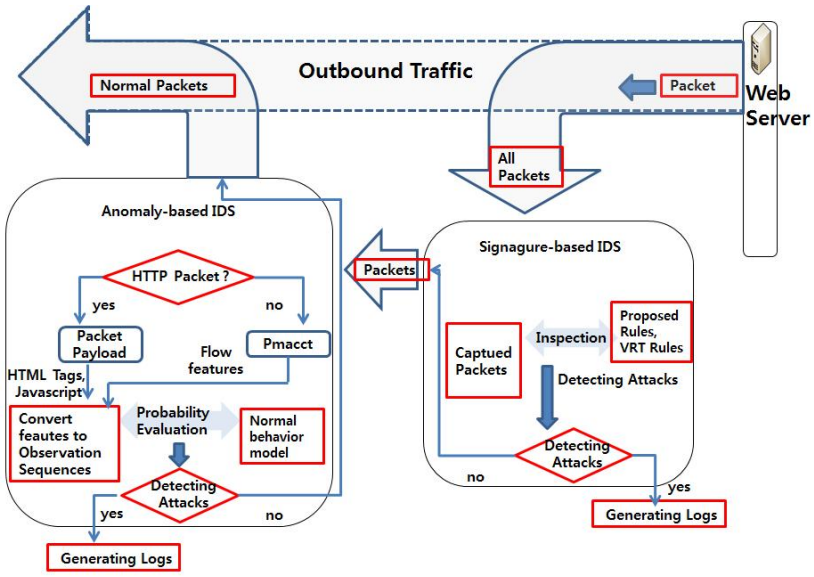

Fig. 3. Proposed Two-Step Detection Scheme

Traditional layered security systems detect intrusion, and do not pay attention to what happens after the intrusion. Unlike outsider attacks, detoured attacks make full use of this point, and bypass the traditional Web-based intrusion path. To protect from detoured attacks which the layered security system cannot detect, we proposed a two-step detection scheme. Detoured attacks show abnormal symptoms when packets are sent through the network, as discussed in Section 2. That is, abnormal contents of HTTP packets are transferred outwards through the network or outbound non-HTTP packets show abnormal traffic features. Therefore, instead of analyzing user's activities as done in the insider detection system, and instead of inspecting inbound packets as done in a traditional security system, we propose to inspect and detect abnormal outbound traffic caused by the detoured attack as shown in Figure 3.

3.2. Signature-based Detection

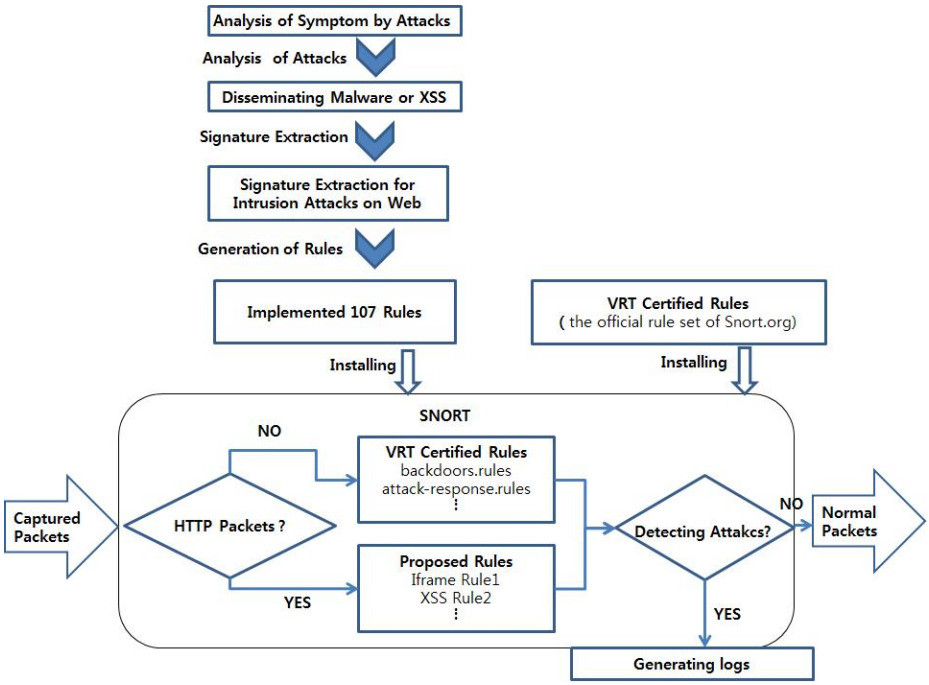

Fig. 4. Process of signature-based detection

The first step of our detection scheme is signature-based detection, which we implement using Snort which is an open source network intrusion prevention and detection system that combines the benefits of signature, protocol,

and anomaly-based inspection. A signature is a distinctive mark or characteristics contained in the context of a packet. Our signature-based IDS detects the symptoms of disseminating malware, XSS, URL Spoofing, information leakage and other abnormality from the Web server.

All these symptoms are represented as special forms of Tags and JavaScript codes as well as particular context in outbound packets. Thus, the above attacks could be detected by finding predefined signatures in the HTML documents transferred from the Web server.

Snort has a function to detect abnormal context in the outbound packets, if proper signatures and rules are provided. Thus, we define signatures found in abnormal HTTP packets with port number 80. We create new rules in Snort to inspect HTTP packets and detect predefined signatures. Our rules detect the actual vulnerability with signatures extracted in abnormal HTML documents. In addition, we use preexisting rules in Snort, called VRT certified rules, to detect abnormal non-HTTP packets generated by backdoor attacks. Figure 4 shows the detailed process of generating and applying rules in Snort from the signatures independent of other attributes.

3.3. Anomaly-based Detection

The second step of our detection scheme is anomaly-based detection which detects attacks by using HMM, and finding the probability of an observed sequence in a normal model. HMM is a statistical model of a system as a Markovian process with hidden states. HMM is characterized by the number of states N, the number of distinct observation symbols per state M, the state transition probability distribution A, the observation symbol probability distribution in a state B, and the initial state distribution π. Given appropriate values of N, M, A, B, and π, HMM can generate an observation sequence O. Thus, HMM requires specifications of two model parameters (N, M), observation symbols, and three probability measurements (A, B, π) and the compact notation λ=(π, A, B) is used to indicate the HMM model[19] . As an application of HMM to detecting attacks, we can find how to compute P(O| λ) under the given model λ=(π, A, B) with observation sequence O, and how to adjust the model parameters λ=(π, A, B) to maximize P(O| λ). With the Baum-Welch algorithm for this problem, we can train with the normal dataset in the same way of finding optimal values for π, A and B to maximize the probability of an observation sequence O given λ. Then, the probability evaluation, which finds the probability of the observation sequences(generated from tags/Javascript codes or flow features in outbound traffic) in the normal model, is used to detect attacks. We have already proposed an IDS with HMM[8].

Our proposed anomaly-based detection checks two events: 1) whether Tags or JavaScript codes in the HTTP outbound packet are normal, and 2) whether non-HTTP packets have abnormal flow features. This process requires the six phases shown in Figure 5. Normal behavior models are created according to

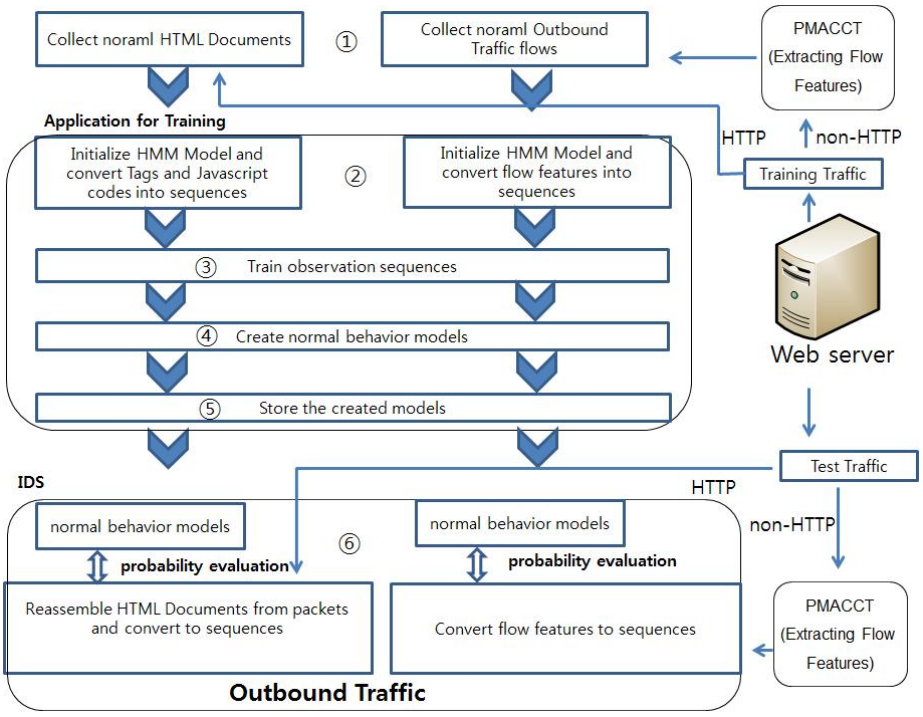

Fig. 5. Process of anomaly-based detection

the training results from both Tags or JavaScript codes in HTTP packets and statistical analysis of non-HTTP packets.

For HTTP packets, tags and JavaScript codes in each packet are applied to probability evaluation. For non-HTTP packets, each packet is monitored by the pmacct which is a small set of passive network monitoring tools to measure, account, classify, aggregate and export IPv4 and IPv6 traffic[25]. Flow features provided by the pmacct are applied to probability evaluation. The selected flow features in the detection are destination IP address, destination port number, source port number, TCP flags, average number of packets per flow, average number of bytes per flow, and time duration between two flows.

Both Tags or JavaScript codes in each HTTP outbound packet and selected features of each flow, are applied to probability evaluation, and attacks are detected according to the normal models that are configured as shown in Figure 5.

4. Verification Analysis

4.1. Test Environment

Table 2 shows a detailed description of our test environment.

4.2. Datasets

We use the following three datasets for the verification of detecting abnormal HTTP packets:

– Dataset1 with 670 Normal HTML documents generated by the common Web server with DB installed in the test environment. This dataset is used to evaluate the False Positive rate.

– Dataset2 with 100 altered HTML documents provided by one of the Korean Security Agencies. They are generated by real inside attacks and detoured attacks. They include obfuscated JavaScript codes or HTML tags, which work in their attempt to download and install malware or adware. This dataset is used to evaluate the detection rate.

– Dataset3 generated in real time by HDSI which is an SQL Injection Hacking Tool. This dataset is used to show detection efficiency.

The following dataset is used as non-HTTP outbound traffic:

– Dataset4 with packets generated by a Web server, in which 10 backdoor programs ($TrojanDropperBackdoorSpyware, GOV-bundestrojaner, APT-RTLO, Crime_Kelihos.B, Crime_Sinowal-Mebroot-Torpigandavariant malware, A16977E9CCBF86168CE20DFC33E0A93C, BBBreport, prorat, trojanSiscosBackdoor-as-FlashInstaller$) are installed. This dataset is

Table 2. Test Environment

device	item	description
Web server for dataset3	DBMS	MS - SQL 2000
	Server	IIS 5.0
	Web Programming Language	ASP
	Virtual machine	MS Virtual PC
Web server for other datasets	DBMS	MS - SQL 2000
	Server	IIS 7.5
	Web Programming Language	ASP
	Virtual machine	MS Virtual PC
Signature-based IDS	IDS	Snort 2.8.6.1
	Packet capture Library	winpcap 4.0
	Virtual machine	MS Virtual PC
Anomaly-based IDS	Packet capture Library	Jpcap 0.7
	HTML Parser	Jericho HTML Parser
	JavaScript Parser	Rhino 1.7 R3
	HMM Library	JaHMM
	JDK (language)	Oracle JDK 1.6 (Java)
	Virtual Machine	MS Virtual PC
	Flow Monitoring Tool	pmacct

used to show the efficiency in detecting abnormal non-HTTP packets generated by backdoor attacks.

4.3. Verification

The performance of the proposed scheme is analyzed in terms of the FP rate, detection rate, and detection efficiency.

Table 3. False Positive rate of each detection scheme

	normal documents	FPs	FP rate (%)
signature-based IDS	670	0	0.000
anomaly-based IDS		3	0.0044

To verify the FP rate, 670 normal HTML documents are randomly requested to the Web server. Table 3 shows the result. The FP rate of signature-based detection is negligible; it shows no error for 670 documents. The signature-based detector checks each packet in the outbound traffic, and classifies the packet as ”attacked” only if any defined signature equals any part of the payload of the packet. Thus, the FP rate, which means the evaluated result is ”attacked” for the unattacked packet, must be negligible. However, the FP of anomaly-based detection happens because of untrained abnormality caused by programming errors and exceptional payment documents.

Table 4. Detection rate of each detection scheme

	normal documents	detection rate
signature-based IDS	100 HTML documents	73%
anomaly-based IDS		89%

Table 5. Detection rate of two-step detection scheme

altered documents	passed documents after 1st step	passed documents after 2nd step	detection rate (%)
100	27 (73 are detected)	2 (98 are detected)	98

To verify the detection rate, we use dataset2 of 100 altered HTML documents. First, the two detection schemes are tested individually. As shown in Table 4, the detection rate of any single scheme is below 90%. Then, we test the same dataset in two steps; first signature-based detection and then anomaly-based detection. The evaluated results show that almost all altered documents generated by the attacks are detected through the two steps and the detection rate is 98%, as shown in Table 5.

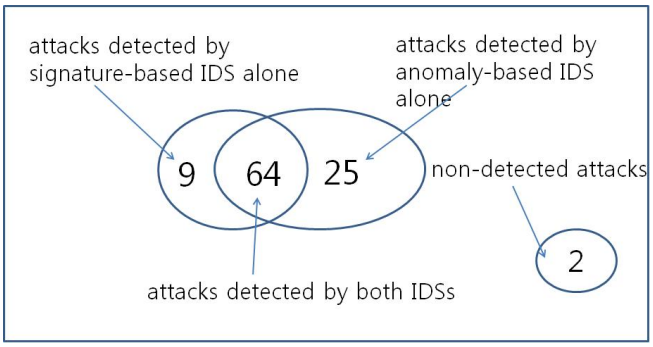

Fig. 6. Venn Diagram of detected attacks

Figure 6 shows how two detection methods in our scheme are complimentary to each other. The signature-based detector cannot detect any unknown or new attacks, but the anomaly-based detector can detect them if they have the same abnormalities as the known attacks have. In Figure 6, 25 detections by anomaly-based IDS alone represent these attacks. On the other hand, the anomaly-based IDS cannot detect any attacks producing untrained abnormality, or indistinctive tags or normal traffic features but the signature-based detector can detect them if they have special context in the outbound packets. 9 detections by signature-based IDS alone represent these attacks. Thus, our proposed two-step detection scheme increases the detection rate by detecting both unknown and untrained attacks, and the evaluated result agrees with the proposal that the two detection methods are complementary to each other.

Table 6. Detection of attacks generated by HDSI

Stage	Signature-based IDS	Anomaly-based IDS
1st stage	none	19
2nd stage	none	14
3rd stage	none	86
4th stage	none	311
5th stage	30	90

As a test of the detection efficiency, we use dataset3 generated by inputting various SQL queries as Web parameters in HDSI. This test is performed under the most vulnerable Windows 2000. HDSI tries to attack a DB in the Web server through 5 stages. Each stage generates attacks from the Web server, in order to get detailed data. An anomaly-based detector detects abnormal documents in each stage: 19 anomalies in the 1st stage, 14 anomalies in the 2nd stage, 86 anomalies in the 3rd stage, 311 anomalies in the 4th stage and finally 90 anomalies (3 anomalies per each e-mail, 30 e-mails) in the 5th stage, as shown in Table 6. From the analysis result, we can find the efficiency of anomaly-based detection for different anomalies caused by various attacks.

On the other hand, the signature-based detector detects the 30 e-mail attacks in the 5th stage only. This is because error messages only are generated in the 1st - 4th stages and no signatures are found in them. Only e-mails disclosed in the 5th stage have predefined signatures.

For the evaluation of the detection of backdoor attacks, we install 10 backdoor programs to remotely access the Web server. Then, abnormal traffic patterns in non-HTTP packets generated by the backdoor attacks are inspected.

Table 7. FP rate of abnormal non-HTTP packets

	Normal Flows	FPs	FP rate (%)
Signature-based IDS	926 Flows	0	0
Anomaly-based IDS		17	1.83

Table 8. Detection rate of abnormal non-HTTP packets

Malicious Outbound flows	After 1st step (Detection rate)	After 2nd step (Detection rate)
443	22 (4.9%)	443 (100%)

As shown in Table 7, the FP rate of signature-based detection is 0%, because of the same reason as in Table 3. However, the untrained traffic features in normal flows cause the FP rate of anomaly-based detection as high as 1.83%. Thus, we need more efficient traffic features to reduce the FP rate. From the detection rate shown in Table 8, our two-step scheme detects all abnormal non-HTTP flows. However, the detection rate of signature-based detection in the first step is relatively low at 4.9%. Thus, we need more effective signatures to increase this rate.

5. Summary

Even though external outsider attacks are constantly evolving and increasing, they are well detected and protected with the corresponding technical improvement. Detoured attacks (including insider attacks,) unusual type of attacks, become more serious, and pose a common threat that bypasses the traditional Web-based attack path for malicious purpose. It is found that many detoured attacks to the Web server disclose confidential/private information or disseminate malware codes through the Web, and this harm could be protected from, by inspecting their outbound traffic.

In this paper, we propose an improved detection scheme for detoured attacks by inspecting outbound traffic based on the analysis addressed in the previous sections. Our proposed scheme has a two-step hierarchy; the first step is signature-based detection using Snort and the second step is anomaly-based detection using HMM. To detect both abnormal Tags/JavaScript codes and abnormality caused by non-HTTP traffic, the second step inspects both payload and traffic features of the packets for probabilistic evaluation in HMM. We cannot find any other hybrid system that combines two approaches to detect detoured attacks.

Through the verification analysis under real attacked Web server environments, it has been shown that the proposed scheme causes a satisfactory false positive rate and detection efficiency for attacks to the Web server. In particular, the evaluated result agrees with the analysis that the two detection methods in our scheme are complementary to each other for detecting altered HTTP packets as shown in Table 9. In addition, our anomaly-based detector shows good efficiency in detecting abnormal non-HTTP packets caused by backdoor attacks. Our work on detecting abnormal non-HTTP packets is in progress; we need to supplement more signatures and traffic features for complete two-step detection as shown in the verification analysis.

Table 9. Comparison of two detection methods for abnormal HTTP packets

Features	Signature-based	Anomaly-based
Weakness	unknown attack	untrained abnormality
Method	signature matching	HMM
Complexity	simple	complex (time consuming)
FP rate	almost none	very low
Detection rate	relatively low	very high

As an extension, our proposed scheme may be applicable to client-side hosts and mobile devices. In particular, mobile malware has emerged as a serious threat to resource-constrained handheld devices over the past few years. For example, outbound traffic from a smartphone becomes more dangerous, due to confidential information leakage, the scanning of private documents and DDOS attack.

Acknowledgments. The research was conducted by the research fund of Dankook University in 2013.

References

1. The WASC Threat Classification v2.0. Tech. rep., The Web Application Security Consortium (2010), http://projects.webappsec.org/w/page/13246978/Threat%20Classification
2. Aydn, M.A., Zaim, A.H., Ceylan, K.G.: A hybrid intrusion detection system design for computer network security. Computers & Electrical Engineering 35(3), 517 – 526 (2009), http://www.sciencedirect.com/science/article/pii/S0045790609000020
3. Borders, K., Prakash, A.: Quantifying information leaks in outbound web traffic. In: Proceedings of the 2009 30th IEEE Symposium on Security and Privacy. pp. 129–140. SP '09, IEEE Computer Society, Washington, DC, USA (2009), http://dx.doi.org/10.1109/SP.2009.9
4. Bowen, B.M., Hershkop, S., Keromytis, A.D., Stolfo, S.J.: Baiting inside attackers using decoy documents
5. Cho, S.B., Han, S.J.: Two sophisticated techniques to improve hmm-based intrusion detection systems. In: RAID. pp. 207–219 (2003)
6. Choi, B., Cho, K.: Detection of insider attacks to the web server(mist 2012 volume 1). Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA) 3(4), 35–45 (12 2012)
7. Choi, B., Choi, S., Cho, K.: An Efficient Detection Scheme of Web-based Attacks through monitoring HTTP Outbound Traffics. Journal of The Korea Society of Computer and Information 16(1), 125–132 (2011)
8. Choi, B., Choi, S., Cho, K.: Anomaly Detection Scheme of Web-based Attacks by applying HMM to HTTP Outbound Traffic. Journal of The Korea Society of Computer and Information 17(5), 33–40 (2012)
9. Desmet, L., Piessens, F., Joosen, W., Verbaeten, P.: Bridging the Gap Between Web Application Firewalls and Web Applications. In: Proceedings of the 2006 ACM Workshop on Formal Methods in Security Engineering. pp. 67–77 (2006)
10. Ding, Y.X., Xiao, M., Liu, A.W.: Research and implementation on snort-based hybrid intrusion detection system. In: Machine Learning and Cybernetics, 2009 International Conference on. vol. 3, pp. 1414 –1418 (july 2009)
11. Gómez, J., Gil, C., Padilla, N., Baños, R., Jiménez, C.: Design of a snort-based hybrid intrusion detection system. In: Proceedings of the 10th International Work-Conference on Artificial Neural Networks: Part II: Distributed Computing, Artificial Intelligence, Bioinformatics, Soft Computing, and Ambient Assisted Living. pp. 515–522. IWANN '09, Springer-Verlag, Berlin, Heidelberg (2009), http://dx.doi.org/10.1007/978-3-642-02481-8_75
12. Jovicic, B., Simic, D.: Common web application attack types and security using asp.net. Comput. Sci. Inf. Syst. 3(2), 83–96 (2006), http://dblp.uni-trier.de/db/journals/comsis/comsis3.html#JovicicS06
13. Justin Crist: Web Based Attacks. Tech. rep., SANS Institute (2010), http://www.sans.org/reading_room/whitepapers/application/web-based-attacks_2053
14. K., S., P., H.: Guidelines on Firewalls and Firewall Policy. Tech. rep., Computer Security Division Information Technology Laboratory National Institute of Standards and Technology Gaithersburg (2009)

15. Khreich, W., Granger, E., Sabourin, R., Miri, A.: Combining hidden markov models for improved anomaly detection. In: Proceedings of the 2009 IEEE international conference on Communications. pp. 965–970. ICC'09, IEEE Press, Piscataway, NJ, USA (2009), http://dl.acm.org/citation.cfm?id=1817271.1817451
16. Maloof, M.A., Stephens, G.D.: Elicit: a system for detecting insiders who violate need-to-know. In: Proceedings of the 10th international conference on Recent advances in intrusion detection. pp. 146–166. RAID'07, Berlin, Heidelberg (2007), http://dl.acm.org/citation.cfm?id=1776434.1776446
17. Mike Boberski, Juan Carlos Calderon et al.: OWASP Top 10 - The Ten Most Critical Web Application Security Risks. Tech. rep., OWASP (2010), https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project
18. Pontarelli, S., Greco, C., Nobile, E., Teofili, S., Bianchi, G.: Exploiting dynamic reconfiguration for fpga based network intrusion detection systems. In: Field Programmable Logic and Applications (FPL), 2010 International Conference on. pp. 10 –14 (31 2010-sept 2 2010)
19. Rabiner, L.R.: A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. In: Proceedings of the IEEE. pp. 257–286 (1989)
20. Rossow, C., Dietrich, C.J., Bos, H., Cavallaro, L., van Steen, M., Freiling, F.C., Pohlmann, N.: Sandnet: network traffic analysis of malicious software. In: Proceedings of the First Workshop on Building Analysis Datasets and Gathering Experience Returns for Security. pp. 78–88. BADGERS '11, ACM, New York, NY, USA (2011), http://doi.acm.org/10.1145/1978672.1978682
21. Salem, M., Hershkop, S., Stolfo, S.: A Survey of Insider Attack Detection Research. In: Stolfo, S., Bellovin, S., Keromytis, A., Hershkop, S., Smith, S., Sinclair, S. (eds.) Insider Attack and Cyber Security, Advances in Information Security, vol. 39, pp. 69–90. Springer US (2008), http://dx.doi.org/10.1007/978-0-387-77322-3_5
22. Salimi, E., Arastouie, N.: Backdoor detection system using artificial neural network and genetic algorithm. In: Proceedings of the 2011 International Conference on Computational and Information Sciences. pp. 817–820. ICCIS '11, IEEE Computer Society, Washington, DC, USA (2011), http://dx.doi.org/10.1109/ICCIS.2011.103
23. Shaimaa Ezzat Salama, Mohamed I. Marie, L.M.E.F., Helmy, Y.K.: Web Anomaly Misuse Intrusion Detection Framework for SQL Injection Detection. International Journal of Advanced Computer Science and Applications 3(3), 123–129 (2012)
24. Skrzewski, M.: Analyzing outbound network traffic. In: Kwiecie, A., Gaj, P., Stera, P. (eds.) Computer Networks, Communications in Computer and Information Science, vol. 160, pp. 204–213. Springer Berlin Heidelberg (2011), http://dx.doi.org/10.1007/978-3-642-21771-5_22
25. pmacct project team: pmacct project, http://www.pmacct.net/
26. Wang, X., Luo, J., Yang, M., Ling, Z.: A potential http-based application-level attack against tor. Future Generation Computer Systems 27(1), 67 – 77 (2011), http://www.sciencedirect.com/science/article/pii/S0167739X10000713
27. Yim, K., Hori, Y.: Information leakage prevention in emerging technologies (mist 2012 volume 2). Journal of Internet Services and Information Security (JISIS) 2(3/4), 1–2 (2012)

Byungha Choi received the MS degree from the Dept. of Information and Communication Technology, Dankook University. He is currently a Ph.D. student of Dept. of Computer Science and Engineering at Dankook University. His research interest is Network Security.

Kyungsan Cho(Corresponding Author) received his B.Sc. in Electronics Engineering(Seoul National University, 1979), master degree in Electrical and Electronic Engineering(KAIST, 1981), and his Ph.D. degree in Electrical and Computer Engineering(the University of Texas at Austin, 1988). During 1988-1990, he served as a senior R&D engineer at Samsung Electronics Company. He joined Dankook University in March 1990, where he is currently a professor in the department of software science. He authored several books in Computer Architecture and Computer Networks and published over 40 academic papers. His research interests include mobile networks, network security and traffic analysis.

Received: September 8, 2012; Accepted: March 11, 2013.