International Journal of Internet, Broadcasting and Communication Vol.15 No.3 149-159 (2023)
http://dx.doi.org/10.7236/IJIBC.2023.15.3.149

IJIBC 23-3-19

A Study on Scenario-based Web Application Security Education Method

Gilja So

Professor, Department of Cyber Security Youngsan University, Korea
kjso@ysu.ac.kr

Abstract

Web application security education that can provide practical experience is needed to reduce damage caused by the recent increase in web application vulnerabilities and to strengthen security. In this paper, we proposed a scenario-based web application education method, applied the proposed method to classes, and analyzed the results. In order to increase the effectiveness of scenario-based education, a real-life practice environment to perform scenarios and instructions to be performed by learners are needed. As an example of the proposed method, instructions to be performed by learners from the viewpoint of the attacker and the victim were shown in a practice environment to teach XSS and SQL injection vulnerabilities. After applying the proposed method to the class for students majoring in cyber security, when the lecture evaluation results were analyzed, it was shown that the learner's interest, understanding, and major ability all improved.

Keywords: *Web application vulnerability, scenario-based education, SQL injection, XSS attack, Information Security .*

1. Introduction

Recently, with the advancement of the internet environment, web-based application services are being provided in various fields, but damage incidents exploiting web application security vulnerabilities are also increasing. According to a report published in 2020 by ENISA(The European Union Agency for Cybersecurity) , web-related attacks ranked second and fourth in the ranking of cyber threats [1].

Organizations such as OWASP (Open Worldwide Application Security Project)[2], WASC (Web Application Security Consortium)[3], and SANS[4] provide guidelines for diagnosing and defending web application vulnerabilities, and in Korea, KISA (Korea Internet & Security Agency)[5] distributes “web server and homepage vulnerability inspection guide” to minimize damage caused by web vulnerabilities. In order to maintain web application security, various stakeholders such as developers, operators, and users must have sufficient understanding and knowledge about security. To this end, web application security education is essential, and strategies and methods are required to increase the effectiveness of education. Research topics on information security education are largely researched into curriculum, education method, education system, competency development, educational content, and effectiveness measurement. However, when comparing domestic and foreign research topics, research related to educational content development is relatively

Manuscript Received: June. 25, 2023 / Revised: July. 2, 2023 / Accepted: July. 6, 2023
Corresponding Author: kjso@ysu.ac.kr
Tel: +82-55-380-9530, Fax: +82-55-380-9249
Professor, Department of Cyber Security, Youngsan University, Korea

insufficient compared to other topics, with 35% research on curriculum and 7.4% research on educational content development [6]. In particular, since web security education is necessary not only for security majors but also for operators and users, educational contents that can enhance learners' understanding and arouse interest are needed.

In web security education, practical experience is a very effective method, so there are studies to develop web security education tools to provide practical experience [7, 8]. [7] proposed SWEET, a tool for educating web application security development for undergraduate and graduate students. [8] analyzed the curriculum for the security department of Meister and specialized high schools in Korea and designed practical educational software that would be helpful in the classroom. Web security education tools are configured to understand the causes of vulnerabilities and diagnosis methods for overall web security education, but they do not provide actual attack scenarios. According to research of [9], scenario-based security education in network security practice classes shows significant results in arousing learners' interest. In web application security education, by creating and using exploits to attack target servers or users according to attack scenarios, learners' interest in web security can be increased, and web vulnerability diagnosis and defense capabilities can be improved.

This paper proposes a scenario-based web security education model as a strategy and method to increase the effectiveness of web application security education. For scenario-based education, practice environments suitable for each scenario, attack scenarios, and instructions to be performed by learners are required. When configuring the practice environment, the web server was composed of a public website to test web application security vulnerabilities or a server built locally. Attack scenarios and execution instructions for each scenario are configured step by step so that learners can learn easily. The model proposed in this paper was applied to cyber security major student classes, and the results of class satisfaction show that scenario-based learning increases students' interest and improves their major competency.

The structure of this paper is as follows. Chapter 2 explains the guidelines and methods for diagnosing web application security vulnerabilities necessary for security education, and explains the recent research trends related to security education contents. Chapter 3 explains the scenario-based web application security training method with examples of XSS and SQL Injection vulnerabilities. Chapter 4 analyzes the effectiveness of scenario-based education by using the results of the two-year lecture satisfaction survey. We come to conclusion in Chapter 5

2. Related Works

In this chapter, we explain web application security vulnerability diagnosis guides and methods needed for security education, and recent research trends related to security education contents.

2.1 Web Application Security Vulnerability Inspection Criteria and Diagnosis

Web application security vulnerabilities are often caused by vulnerable codes in the development stage, so they are in line with the secure coding inspection standards. Web application and secure coding inspection standards are provided by organizations such as OWASP[2], WASC[3], and SANS[4]. OWASP provides OWASP Top 10 documents every three years by organizing 10 representative vulnerabilities that occur on the web. CWE/SANS top 25 is a standard distributed by SANS, an American security education institute. In Korea, the “Web Server and Homepage Vulnerability Check Guide” is distributed by referring to the CWE inspection standards by KISA (Korea Internet & Security Agency) [5].

There are methods such as manual diagnosis by an external expert, use of an automated inspection tool, and source code diagnosis to diagnose web application vulnerabilities. The automated inspection tool

automatically analyzes vulnerabilities using a web scanner, which automatically collects the structure of the site and analyzes vulnerabilities. However, automatic inspection is difficult to accurately detect, so it must be combined with manual detection by external experts. Therefore, the web security curriculum must include a method for manually detecting each vulnerability [10].

2.2 The Researches for the Development of Contents for Cyber Security Education

In security education, practical experience is a very effective education method, so there are studies to develop various contents to provide practical experience. [7], [8], [9], [11], [12], [13] are researches on information security education contents development. [11] raises the issue that while hacking challenges are helpful in educating students on security, they are time consuming to develop and are static once created, so that once a problem is solved, students have no more problems to challenge. To solve this problem, a virtual environment is built according to a random scenario, and CTF-events and hints are provided in the built virtual environment. [9] proposed a virtualization environment and scenario for network security practice for students of specialized high schools. Network security training was conducted by configuring a virtualization environment similar to the actual service environment and providing practice scenarios that can be practiced in the virtualization environment. Pre and post tests were conducted to measure creative problem-solving ability and learning motivation for the experimental group, and it was confirmed that providing a virtual environment and training scenarios had significant results in learning motivation for network security learning. [12] converts the physical devices of the smart factory into virtual machines or simulation models for security education of the smart factory, configures a digital twin practice environment, and creates a scenario that damages virtual elements in the practice environment by generating ransomware. [13] proposed five scenarios in which an attack could occur by organizing a working team, operation team, attack team, and defense team as a suitable cyber hacking training method in the IIOT/CPS environment. Each scenario proposes 5 scenarios in which an attack can occur under the assumption that an attacker can exist anywhere on the 4 teams.

As studies related to web application security education contents, [7] and [8] proposed tool-based education contents. [7] proposed SWEET, a tool for educating web application security development for undergraduate and graduate students. SWEET was developed with the goal of providing a learning experience for web application security development through a standardized computing environment and learning module. Eight teaching modules were proposed, and lecture materials and hands-on exercises are provided for each teaching module. Among them, the web application stress testing module enables penetration testing for SQL injection, XSS attack, and poor authentication by using a web server with vulnerabilities. As a server with vulnerabilities, WebGoat[14] developed by OWASP is used to explain the causes of vulnerabilities and how to infiltrate them. [8] analyzed the curriculum for the security department of Meister and specialized high schools in Korea and designed practical educational software that would be helpful in the classroom. It consists of step-by-step exercises to identify attacks and principles, and is configured to practice SQL, XSS, CSRF, upload download, operating system command execution vulnerabilities, etc. [7] and [8] are structured to understand the causes of vulnerabilities and diagnosis methods for overall web security education, but they do not provide practical attack scenarios.

3. Scenario of Web Vulnerability Education

In the OWASP Top 10, injection vulnerabilities are classified as high-risk vulnerabilities. In this study, scenarios for security education on XSS and SQL injection vulnerabilities are presented as examples.

3.1 Cross-Site Scripting (XSS)

3.1.1 Overview of Cross-Site Scripting (XSS) attack

Web applications use various scripts to improve user interaction, and web browsers execute these scripts on the user's PC. XSS is an attack that exploits this point to cause damage to users accessing web applications. Through an XSS attack, an attacker can steal the victim's cookie values, passwords, credit card information, etc. As user interaction becomes more important in the web application environment, XSS vulnerabilities can become a risk factor. XSS attacks are largely classified into stored XSS, reflected XSS, and DOM-based XSS. Stored XSS attack is when an attacker inputs a contaminated script that is stored in the database and exposed to other users. The reflected XSS attack is a process in which a contaminated script is replayed on the screen and the script is executed on the user's PC. DOM-based XSS can be attacked in either reflection or stored form. DOM stands for Document Object Model and is a programming interface that expresses the structure of web documents. Elements composing web pages are treated as objects, and objects can be manipulated through scripting languages such as JavaScript. If the user's input value is applied to the DOM object as it is, the DOM object can be polluted and damage occurs when a script using this object is executed. Stored XSS is an easy-to-understand attack method from the developer's point of view, since malicious scripts are stored in the database and damage occurs when the content is accessed. However, reflected XSS is an attack method that directly damages the victim's browser without the malicious script being stored on the server or hitting the server. When a learner learns reflected XSS for the first time, it is very easy to diagnose vulnerabilities, but it is not easy to understand the attack mechanism and how an attack is possible using it. Therefore, in the case of reflected XSS, it is possible to better diagnose and respond to vulnerabilities when learning through attack scenarios.

3.1.2 The scenario of reflected XSS attack

Figure 1 illustrates the scenario of reflexed XSS attack. Followings are steps of exploiting the vulnerability of the reflected XSS and the victim transmits his or her sensitive information the attacker's server in figure 1.

Step 1. The attacker sends a phishing e-mail to the victim by including a malicious script that creates a login form in the URL of the website with the reflected XSS vulnerability.

Step 2. The victim requests the URL containing the malicious code of the phishing email to the trusted server.

Step 3. The server responds to the victim by including malware that creates a login form in the message.

Step 4. The victim's web browser executes the malicious code to create a login form.

Step 5. The victim sends his identification and password to be used to login the trusted server to the attacker's server through the login form created by the attacker.

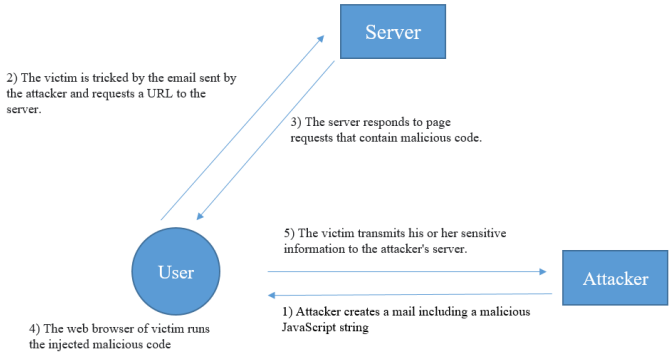
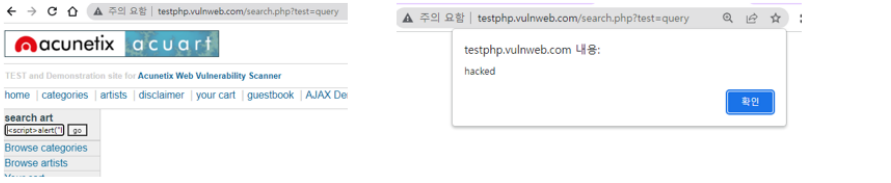


Figure 1. The scenario of phishing attack using reflected XSS vulnerability

3.1.3 Environment for Hand-on Experience

(1) Servers with Reflected XSS Vulnerability

A website (http://testphp.vulnweb.com) is created for the purpose of web scanner testing by Acunetix [15]. This website has a reflected XSS vulnerability in its search function. Figure 2 shows the result of diagnosing the server's reflected XSS vulnerability. A user supply search term in a URL parameter of search function with JavaScript such as "<script>alert('hacked')</script>" , showed in figure 2 (a), then the user-supplied script is executed in the user's browser, showed in figure 2 (b).



(a) JavaScript code supplied for search term (b) Execution of user-supplied JavaScript code

Figure 2. Diagnosing Server's Reflected XSS Vulnerability

(2) Login form to be sent to victims with malware

Figure 3 illustrates the login form and HTML source of the server to be included in the malicious phishing email to be sent to the victim. In a normal login operation, the user's input is sent to the server, but in the login form recreated by the attacker, the user's login information is sent to the attacker.

(a) Login form of trusted server (b) HTML source of login form of server

Figure 3. Login form and HTML source code of server

3.1.4 Instructions of phishing attack using reflected XSS vulnerability

Table 1 is a step-by-step instructions to perform an attack scenario in which an attacker steals user's personal information by creating a phishing email using the URL of a trusted server with a reflected XSS vulnerability. The victim provides his/her personal information without question because the accessed URL is the one of the trusted server, but the URL provided by the phishing email contains malicious code and transmits the victim's personal information to the attacker's server.

Table 1. Instructions of phishing attack

Actor		Instructions
Attacker	Step 1	Check that target server has the reflected XSS vulnerability After providing <script>alert("hacked")</script> in the search box, check if it is executed.

	Step 2	Writing a phishing email containing malicious code like following messages. Today you will receive 1000 bonus points redeemable for cash when you log in to this site. Don't miss the great opportunity and come and visit us Go to login
	Step 3	Create malicious code that will be executed while connecting to the server when “Go go login” is clicked. Malicious code creates login form of trusted server and transmits user’s input data to attacker's server. In this case, the URL of attacker’s server is http://localhost:8080/XssAttack/insertid.jsp. <script>document.getElementById(₩'pageName₩').innerHTML= ₩'<div id=₩"content₩"><div class=₩"story₩"> If you are already registered please enter your login information below: <form action=http://localhost:8080/XssAttack/insertid.jsp> <table cellpadding=₩"4₩" cellspacing=₩"1₩"><tr><td>Username : <₩/td><td><input name=₩"id₩" type=₩"text₩" size=₩"20₩" style=₩"width:120px;₩"><₩/td><₩/tr><tr><td>Password : <₩/td><td><input name=₩"pw₩" type=₩"password₩" size=₩"20₩" style=₩"width:120px;₩"><₩/td><₩/tr><tr><td colspan=₩"2₩" align=₩"right₩"><input type=₩"submit₩" value=₩"login₩" style=₩"width:75px;₩"><₩/td><₩/tr><₩/table>₩' </script>
	Step 4	Send to phishing e-mail to the victim
Victim	Step 1	Click the “Go to login” hypertext of the received mail to connect to the server, enter identification and password, and try to log in.

3.2 SQL Injection Vulnerability

3.2.1 Overview of the SQL Injection Attack

Traditionally, many web applications have been written based on scripting languages such as PHP, JSP, or ASP. The html code and the database processing script were mixed and used on the same page, and the user's input value was directly connected to the script statement using the database query statement without filtering. During this process, the malicious code provided by the attacker modifies the SQL query, and damages such as stealing personal information stored in the server, bypassing login, acquiring authentication, and executing system commands using database extension procedures can occur.

3.2.2 The scenario of the SQL Injection Attack

Figure 4 shows an attack scenario in which the personal information of members who have joined the shopping mall is leaked by using the SQL injection vulnerability. The shopping mall that is the target of the attack has the function of logging in, searching for products, and viewing the product list. In (1), the attacker inserts malicious SQL code to steal the personal information of members registered in the shopping mall and sends it to the shopping mall server. In (2) and (3) of figure 4, the shopping mall server normally executes the SQL query inserted by the attacker, and as a result, in figure 4 (4), the user information of the shopping mall is sent to the attacker.

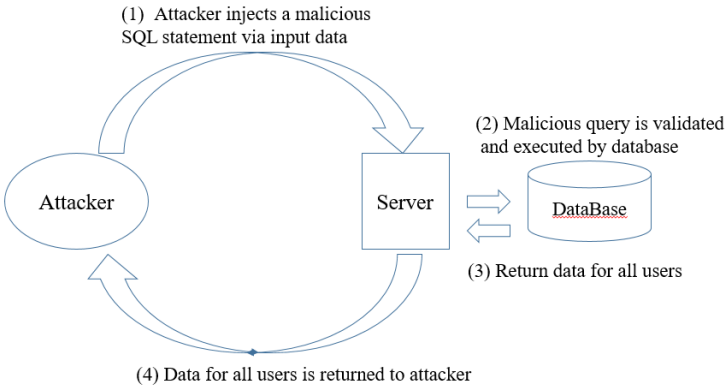


Figure 4. The scenario to dump data for all users from a shopping mall

3.2.3 Environment: A Shopping Malls with SQL Injection Vulnerabilities

(1) Tables Created in Shopping Malls

The attack target is a shopping mall implemented as a web application using a database. There is a web application implemented in PHP to manage product information and member information registered in the shopping mall as a database, and to display a product list. Figure 5 shows a table description for storing product information and member information of a shopping mall in figure 5 (a), figure 5 (b) each.

Field	Type	Null	Key	Default	Extra
goodid	int(11)	NO	PRI	NULL	
name	varchar(200)	YES		NULL	
category	char(1)	YES		NULL	
price1	int(11)	YES		NULL	
content	varchar(1000)	YES		NULL	
image	varchar(50)	YES		NULL	
regdate	date	YES		NULL	

(a) The description of table "goods"

Field	Type	Null	Key	Default	Extra
id	varchar(30)	NO	PRI	NULL	
pwd	varchar(20)	YES		NULL	
name	varchar(100)	YES		NULL	
email	varchar(40)	YES		NULL	
zip_code	varchar(7)	YES		NULL	
address	varchar(100)	YES		NULL	
phone	varchar(20)	YES		NULL	

(b) The description of table "users"

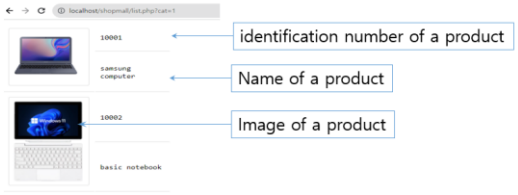
Figure 5. The description of tables created in shopping malls

(2) The Example Codes of Shopping Malls with Vulnerabilities

The code of PHP, a web application that displays the product list information stored in the database table goods on the web screen, is shown Figure 6 (a). Figure 6 (a) shows the execution screen of code in figure 6 (b). The URL format for executing the code in figure 6 (a) is "http://localhost/shopmall/list.php?cat=1". The URL input variable "cat" is a variable that designates the category of the product. The products of the category corresponding to the value set in variable "cat" are retrieved from the database and displayed on the screen. Looking at line 3 of figure 6 (a), there is a SQL injection vulnerability because the external input value is used in the SQL query statement without any filter.

```
1: <?php
2: $category = $_GET['cat']
3: $query= "select goodid,name,image from goods where category ='$category'"
4: mysql_query($query)
```

(a) PHP code to retrieve list of products



(b) Display of URL "http://localhost/shopmall/list.php?cat=1"

Figure 6. The example code with vulnerability and how to use it

3.2.4 Instructions of Retrieving Database of Victim Server

Table 2 shows instructions for attacking shopping mall with SQL injection vulnerabilities and retrieving personal information of all users joined in shopping mall.

Table 2. Instructions for SQL Injection Attack

Actor		Instructions
Attacker	Step 1	Find out the number of fields used in the webpage showing the product list using the method below. *http://localhost/shopmall/list.php?cat=1 union select 1,2,*3 or http://localhost/shopmall/list.php?cat=1 union select 1,2,3,4
	Step 2	Find the DB name used by the shopping mall server using the method below. *http://localhost/shopmall/list.php?cat=1 union select 1,database(),3*
	Step 3	Finds the table names of database used by the shopping mall server using the method below. *http://localhost/shopmall/list.php?cat=1 union select 1,group_concat(table_name),3 from information_schema.tables where table_schema='goods'*
	Step 4	Among the list of tables found in step 3, find the column names of the table that will hold the personal information of the members using the method below. *http://localhost/shopmall/list.php?cat=1 union select, group_concat(column_name), 3 from information_schema.columns where table_name='users'*
	Step 5	Extract member information by using the column name obtained in step 4 using the method below. *http://localhost/shopmall/list.php?cat=1' union select group_concat(name), group_concat(pwd), 3 from users--*

Figure 7 (a) shows the list of members who have joined the DB, and figure 7 (b) shows the result of exposing the list of members in the DB to the web screen by SQL injection attack. As a result of the attack, the personal information of all members registered in the DB can be leaked

(a) Records retrieved from table "goods" (b) Personal information of members leaked by SQL injection attack

Figure 7. The result of SQL injection attack

4. Analysis of the Effectiveness of the Proposed Method

In order to analyze the effectiveness of scenario-based web application security education, the lecture evaluation results of the "Web Hacking and Defense Practice" course opened in the Cyber Security Department of Youngsan University in South Korea are analyzed to teach web application security. Figure 8 shows the course evaluation results in 2022 and 2023.

(a) Course evaluation results for 2022 (b) Course evaluation results for 2023

Figure 8. Course evaluation results for 2022 and 2023

Among the lecture evaluation questionnaire items, the items from which the conclusion of the proposed method can be inferred are interest, understanding, and major ability improvement. Table 3 shows the contents and numbers of survey questions to evaluate interest, understanding, and major ability. In 2022, items related to major competency improvement were composed of knowledge, skills, and attitude and the related question numbers are 10, 11, and 12 in order. They were integrated into one item in 2023 and the related number is 9.

Table 3. Questionnaires representing interest, understanding, and major ability improvement

Item	Contents	number	
		2022	2023
Interest	Classes were conducted to stimulate students' interest in learning	2	2
Understanding	Classes were conducted so that students could understand the contents of the class.	3	3
Major ability improvement	The class helped me improve my knowledge	10	-
	The class helped me improve my skills	11	-
	The class helped me improve my attitude	12	-
	I was able to improve my major skills through classes.	-	9

The five-step evaluation used in the lecture evaluation questionnaire is divided into high, medium, and low, and the results are analyzed. Among the five levels, “very good” and “good” are classified as high, “average” as medium, and “poor” and “very poor” as low. Table 4 shows the results of students' responses in 2022 and 2023. Analyzing the results, the number of students who were interested in the class increased and their understanding of the class content also increased. The percentage of those who thought that their major skills related to web applications had improved also increased.

Table 4. Lecture evaluation results

Item and level \ year		2022	2023	Difference
Interest	H	81%	94%	13%
	M	19%	6%	-13%
	L	0%	0%	0%
Understanding	H	78%	82%	5%
	M	19%	12%	-7%
	L	4%	6%	2%
Major ability improvement	H	78%	94%	16%
	M	22%	6%	-16%
	L	0%	0%	0%

5. Conclusion

To increase the learning effect of web application security education, it is necessary to provide learners with practical experience. Existing web application security education is sufficient to learn how to diagnose the cause of a vulnerability and respond to it, but it does not provide practical experience. In this paper, we proposed a scenario-based web application security education method to solve this problem. In order to learn using the scenario-based web application model, an actual practice environment and instructions from the attacker's or victim's point of view to perform the scenario were provided. In order to analyze the effect of applying the proposed model to students majoring in cyber security, the two-year lecture evaluation results were analyzed in terms of interest, understanding, and major ability improvement. Students' interest in attack scenarios using web application vulnerabilities and practical practice environments increased, and their understanding of web application vulnerabilities also increased by understanding how vulnerabilities can be used by attackers beyond simple vulnerability diagnosis. As a result, the major ability was also improved.

In order to provide more experiences to students in the future, various scenarios, necessary practice environments, attack methods, and execution instructions need to be created. The more scenarios that can be practiced, the more choice learners will be guaranteed and the learning effect will increase.

Acknowledgement

This work was supported by Youngsan University Research Fund of 2022.

References

[1] H.H Jin and H.K Kim, " A Study on Web Vulnerability Risk Assessment Model Based on Attack Results: Focused on Cyber Kill Chain" The Journal of The Korea Institute of Information Security & Cryptology, VOL.31, NO.4, pp.779-791, Aug 2021.
DOI: https://doi.org/10.13089/JKIISC.2021.31.4.779

[2] OWASP(Open Worldwide Application Security) *https://owasp.org/*

[3] WASC(Web Application Security Consortium) *http://www.webappsec.org/*

[4] SANS Institute *https://www.sans.org*

[5] KISA (Korea Internet & Security Agency) *https://www.kisa.or.kr/*

[6] K.W Kim and J.D Kim, "An Analysis of Research Trends in Information Security Education", The Journal of The Korea Institute of Information Security & Cryptology VOL.26, NO.2, pp.489-499, Apr 2016.
DOI: http://dx.doi.org/10.13089/JKIISC.2016.26.2.489

[7] Li-Chiou Chen, Lixin Tao, Xiangdong Li and Chienting Lin, “A Tool for Teaching Web Application Security”, in Proc. 14th Colloquium for Information Systems Security Education, ,Baltimore, Maryland, pp. 17-24, June 7-9, 2010.

[8] Jieun Kwak, "Analysis of Curriculum of Teaching Security Course in Meister and Specialized High School and Design of Educational Software for Practicing Web and Network Security Attack", Thesis. Ewha Womans University, Korea, 2019.

[9] Byunghee Jeong, "The Development of Virtualization Environment and Scenario-based Network Security Practice Model", Thesis, Korea National University of Education Chung-Buk, KOREA, 2021.

[10] Yujae Hong, "A Study on the Improvement of Website Security Vulnerabilities", Thesis, Dankook University, Korea, 2020.

[11] Z. C. Schreuders, T. Shaw, M.Shan-A-Khuda, G. Ravichandran, J.Keighley, and M. Ordean, “Security Scenario Generator (SecGen): A Framework for Generating Randomly Vulnerable Rich-scenario VMs for Learning Computer Security and Hosting CTF Events,” in 2017 USENIX$ Workshop on Advances in Security Education (ASE), 2017.

[12] Suman Nam, Seungmin Lee and Youngsun Park, "Development of Information Security Practice Contents for Ransomware Attacks in Digital Twin-Based Smart Factories", The Journal of The Korea Institute of Information Security & Cryptology, Vol. 31, N0. 5, pp. 1001-1010, Oct 2021.
DOI: https://doi.org/10.13089/JKIISC.2021.31.5.1001

[13] Donghyeok Lee and Namje Park, "Hacking Training Plan for Cyber Security in Industry 4.0", The Journal of KIIT. Vol. 15, No. 5, pp. 47-56, May 2017.
DOI: http://dx.doi.org/10.14801/jkiit.2017.15.5.47

[14] WebGoat *https://owasp.org/www-project-webgoat/*

[15] Acunetix *https://www.acunetix.com/*