Pertanika J. Sci. & Technol. 31 (6): 2973 - 2993 (2023)

PERTANIKA JOURNALS

SCIENCE & TECHNOLOGY

Journal homepage: http://www.pertanika.upm.edu.my/

Measuring Vulnerability Assessment Tools' Performance on the University Web Application

Pita Jarupunphol, Suppachochai Seatun and Wipawan Buathong*

Department of Digital Technology, Phuket Rajabhat University, 83000, Phuket, Thailand

ABSTRACT

This research measured vulnerability assessment tools' performance on a university web application, including Burp Suite and OWASP ZAP. There are three measurement criteria: (1) the number of vulnerabilities classified under risk and confidence metrics, (2) the number of vulnerability types and URL alerts classified under risk and confidence metrics, and (3) the number of vulnerabilities classified in the 2021 OWASP Top 10 vulnerabilities. Results showed that Burp Suite detected more vulnerabilities and alerts than OWASP ZAP, with a higher proportion of high-risk vulnerabilities. However, OWASP ZAP had a higher proportion of medium-confidence vulnerabilities. The comparison also revealed that the vulnerabilities identified by both tools were ranked differently within the OWASP Top 10, and there were variations in risk prioritisation between the tools. Despite these differences, the vulnerability assessment results obtained from these tools are still helpful for the university's security analysts and administration, as mitigating cyber threats to the web application is paramount.

Keywords: Cybersecurity, cyber threats, risks, vulnerability assessment, web application

INTRODUCTION

During the COVID-19 pandemic, most academic Institutions must transform their teaching and learning methods to 100% online. These online teaching and learning methods rely on efficient Internet and network systems to fulfil learning activities, such as learning via online meeting applications, sending and storing teaching materials, submitting assignments, and taking online examinations. In the

ARTICLE INFO

Article history:
Received: 19 October 2022
Accepted: 06 March 2023
Published: 03 October 2023

DOI: https://doi.org/10.47836/pjst.31.6.19

E-mail addresses:
p.jarupunphol@pkru.ac.th (Pita Jarupunphol)
s6281423108@pkru.ac.th (Suppachochai Seatun)
w.buathong@pkru.ac.th (Wipawan Buathong)
* Corresponding author

ISSN: 0128-7680
e-ISSN: 2231-8526

meantime, academic institutions have become more vulnerable to cyber threats evolving together with the advancement of Internet technologies. Schools and colleges worldwide are targeted by cyber threats, e.g., abusive content, denial of service (DoS), fraud or deception, information gathering, intrusion attempts, unauthorised access, alteration of important information, and malicious code (Alexei & Alexei, 2021; Pavlova, 2020). As a result, academic institutions must take steps to prevent and mitigate potential risks of cyber threats.

Vulnerability assessment and penetration testing are two different security assessment techniques, but they are crucial in web application vulnerability assessment due to the increasing cyber threats (Darus et al., 2020; Disawal & Suman, 2021; Malekar & Ghode, 2020; Nagpure & Kurkure, 2017). While web application vulnerability assessment is related to flaw detection and analysis and alerts organisations about vulnerable components in the web application, penetration testing is about vulnerability exploitation attempts to determine the feasibility of cyber threats that can affect organisations. Today, various vulnerability assessment tools are available to detect and analyse potential risks associated with web applications. Utilising vulnerability assessment tools may not comprehensively identify vulnerabilities and may produce false positives (FPs). Furthermore, various tools exhibit variability in the vulnerabilities and false positives they report, with some overlap (Alsaleh et al., 2017; Mburano & Si, 2018). These vulnerability assessment tools are varied from proprietary to open-source, which can detect, analyse, and report website vulnerabilities. In addition, the web vulnerability assessment tools provide detailed reports for system administrators to detect and address security issues before any threats occur, according to Khera et al. (2019). However, comparing web vulnerability assessment tools is still inadequate in the literature (Mburano & Si, 2018).

According to cyber threat statistics in 2021 reported by the Thai Computer Emergency Response Team (https://www.etda.or.th/th/Our-Service/thaicert/stat.aspx), there were 939 intrusion attempts, 841 frauds, 570 intrusions, 540 availability, and 271 malicious codes. The results differ from the statistics of the same threats in 2020, in which there were 145 intrusion attempts, 576 frauds, 173 intrusions, 101 availability, and 687 malicious codes. Since the nature of these cyber attacks on public and private organisations may not differ from that on academic sectors, academic institution web applications in Thailand are also vulnerable to cyber threats.This article conducts a web application vulnerability assessment of a university in the south of Thailand using a proprietary tool 'Burp Suite' and an open-source tool 'OWASP ZAP' based on the hypotheses: (1) the university's web application contains high-risk vulnerabilities that might be susceptible to cyber threats, (2) proprietary and open-source tools provide different vulnerability assessment results but are helpful for threat mitigation, and (3) vulnerabilities identified from both tools can be classified into the 2021 OWASP Top 10 (https://owasp.org/Top10/) vulnerabilities.

LITERATURE REVIEW

Cyber Threats in Education

Malekar and Ghode (2020) state that web applications are vulnerable to complex cyber threats. Academic institutions have become an attractive target for cyber-attacks because of several factors. For example, most academic institutions do not invest in information security infrastructure due to limited budgets from the government. Therefore, outdated information systems can be an easy target for attackers. Furthermore, most academic institutions have online databases that collect and store student and staff information varying from personal to financial information (Ulven & Wangen, 2021). Recently, there have been several security incidents in academic sectors. As a result, many schools and colleges have become victims of cyber threats (Naagas et al., 2018; Rahamathullah & Karthikeyan, 2021). For example, an increase in DDoS primarily targets educational institution information systems (Rahamathullah & Karthikeyan, 2021).

Moreover, ransomware attacks can have severe consequences for academic institution operations due to taking much time to restore critical services. The past academic semesters were impacted by ransomware, causing the loss of students' personal information, the institution's financial history, and related information. For instance, the ransomware incident at the University of Northumbria in England, where attackers ransomed the university information system, ceased the university's internal information systems for several weeks (Muncaster, 2020). In addition, ransomware attacks on US schools in 2020 resulted in more than $6 billion in damages from 77 attacks reported by US educational institutions nationwide (Muncaster, 2021).

Vulnerability Assessment

Vulnerability assessment includes techniques and tools to identify information system vulnerabilities and determine the risk of vulnerabilities and the risk assessment objective (Abdullah, 2020; Malekar & Ghode, 2020). Vulnerability assessment has been applied to different aspects of digital technology. For instance, network security assessment involves evaluating an organisational network infrastructure to identify vulnerabilities in public and private networks that may be exposed to threats (McNab, 2016). This assessment checks for vulnerabilities that may interrupt or affect the availability of network services and ports open to access under the host. Similarly, web application vulnerability assessment evaluates potential cyber threats to an online information system that numerous organisations use communication and public relations tools and online services. Since the web application nature is open to users at all times, attackers can take this opportunity to exploit several web application vulnerabilities (Amankwah, Chen, Kudjo et al., 2020). These vulnerabilities can pose security threats to the organisational assets. The vulnerability assessment can help

maintain web application security aspects such as confidentiality, integrity, and availability and mitigate threats by reducing risks arising from the system. In addition, vulnerability assessment can prioritise the most severe vulnerabilities to avoid exploitation (Vibhandik & Bose, 2015).

Web Application Vulnerability Assessment Tools

Web application vulnerability assessment tools are available in proprietary and open-source software that can automate the vulnerability testing process and operate on different operating systems (Diogenes & Ozkaya, 2018). Performing vulnerability assessment can be manual or automatic. However, most vulnerability assessment tools can automatically scan and analyse vulnerabilities and provide detailed reports to help address vulnerabilities that cyber threats can potentially exploit. In addition, the tools collect details about vulnerabilities in their web databases, facilitating further actions for vulnerability assessment. Most web application vulnerability assessment tools collect information within the web application and scan for vulnerabilities that may be exploited within the web application. Once the tool completes all operations, the same procedures will be repeated to increase the accuracy of randomisation of attacks on web applications.

In particular, the proxy is an integral part of web vulnerability assessment tools, allowing the tools to access web applications. For example, Burp Suite, developed by PortSwigger Co., Ltd., is one of the most widely used proprietary web vulnerability assessment tools (Wear, 2018). Acunetix is another proprietary vulnerability assessment tool with advanced crawling technology to search for vulnerabilities in web applications (Ibrahim & Kant, 2018). On the other hand, OWASP ZAP (https://www.zaproxy.org/) is an open-source web vulnerability assessment tool developed by the Open Web Application Security Foundation (OWASP), a non-profit foundation working on several web security improvement projects. OWASP is widely recognised for its OWASP Top 10 vulnerabilities ranking. For example, SQL Injection, Cross-Site Scripting (XSS), and Cross-Site Request Forgery (CSRF) are well-known web vulnerabilities in the 2021 OWASP Top 10.

Vulnerability Assessment Criteria

Vulnerability assessment covers several criteria to analyse, classify and prioritise vulnerabilities.

Risk Levels. Most vulnerability assessment tools classify vulnerabilities into high-risk, medium-risk, and low-risk (Popov et al., 2016). For example, in information security, high risk refers to vulnerabilities that might expose high levels of threats that can affect security requirements in terms of confidentiality, integrity and availability, making the information system unable to operate. This type of risk requires time and a high level of competence

to address issues and bring the information system back to normal. On the other hand, medium risk refers to vulnerabilities that might expose moderate threats affecting some parts of the information security services, such as confidentiality, integrity, and availability. For example, there may be a partial shutdown to repair security issues requiring time and ability to recover, wherein the information system can continue.

Besides, low risk refers to vulnerabilities that expose low-level threats that might affect minimal parts of the system regarding confidentiality, integrity, and availability but without the extended time and the ability to perform corrective actions. As a result, the information system can continue to operate normally. In some vulnerability risk assessments, however, the risk levels can be further extended to 'critical,' which is more severe than the high level. In addition, some vulnerabilities cannot be categorised into these risk levels but are unignorable. For example, informational risk refers to vulnerabilities not susceptible to high, medium, or low-level threats without directly affecting the information system. However, the attacker may exploit these informational-level vulnerabilities to attack the system.

Confidence Levels. Web application vulnerability assessment tools also provide confidence levels to confirm the identified risk level. These confidence levels are also classified into different degrees, e.g., high, medium, low, certain, firm, and tentative. These confidence levels help security administrators determine and prioritise vulnerabilities categorised into risk levels. In this case, the confidence is comparable to likelihood, a significant indicator for risk identification.

The OWASP Top 10 Web Application Security Risks. In addition to the above criteria, the OWASP Top 10 Web Application Security Risks or the OWASP Top 10 vulnerabilities have been web vulnerability assessment criteria widely acknowledged by researchers and practitioners. The OWASP Top 10 ranks vulnerabilities according to severe web application security risk levels. There have been different versions of the OWASP Top 10. Several web vulnerability research articles have widely discussed the 2017 OWASP Top 10 (https://owasp.org/www-project-top-ten/2017/Top_10). Nevertheless, the 2021 OWASP Top 10, a recently updated version, is still unfamiliar in web application vulnerability research. While the 2017 OWASP Top 10 vulnerabilities were selected based on the likelihood, impact, and exploitability determined by experts, the 2021 OWASP Top 10 vulnerabilities were ranked using the same criteria but based on the use of data, if possible.

Significant vulnerability positions and names change from the 2017 OWASP Top 10 to the 2021 OWASP Top 10 (e.g., sensitive data exposure to cryptographic failures). Furthermore, several vulnerability criteria in the 2017 OWASP Top 10 are amalgamated into a new vulnerability type. For example, XML external entities (XXE) are amalgamated with security misconfiguration in the 2017 OWASP Top 10 to only security misconfiguration

in the 2021 OWASP Top 10. There are three new vulnerability categories in the 2021 OWASP Top 10: insecure design, software and data integrity failures, and Server-Side Request Forgery (SSRF).

Related Works

Several research works have been on different web vulnerability assessment perspectives (Abdullah, 2020; Alsaleh et al., 2017; Darus & Awang, 2020; Disawal & Suman, 2021; Karumba et al., 2016). For example, Khalid et al. (2019) proposed a method to predict legitimate or vulnerable code based on six classifiers on a training set consisting of software metrics and text features. The experiment was conducted on three web applications in which 223 vulnerabilities were identified in PHPMyAdmin, Moodle and Drupal. In addition, Darus and Awang (2020) proposed a web assessment tool, 'SNEAKERZ', that automatically detected and analysed vulnerabilities that may arise from the security loophole in web applications based on three software vulnerability categories, including software defects, software bugs, and software errors. The authors asserted that SNEAKERZ could list web vulnerabilities and propose solutions to address the vulnerabilities. In Disawal and Suman (2021), different vulnerabilities were discovered during the web application development process, and a web application vulnerability assessment should be conducted during the web application development to identify factors affecting the web application security, such as weakness, countermeasure, confidentiality impact, access complexity, and severity level. Amankwah, Chen, Kudjo, and Towey (2020) compared the performance of eight web vulnerability assessment tools, including Acunetix, HP WebInspect, IBM AppScan, OWASP ZAP, Skipfish, Arachni, Vega, and Iron WASP, using two vulnerable web applications. The evaluation was based on multiple evaluation metrics. The results show that commercial and open-source vulnerability assessment tools effectively detect vulnerability.

Several research scholars evaluate the performance of open-source web vulnerability assessment tools (Abdullah, 2020; Alsaleh et al., 2017; Amankwah, Chen, Kudjo, & Towey 2020; Karumba et al., 2016; Mburano & Si, 2018). For example, Karumba et al. (2016) introduced a hybrid algorithm for detecting web application vulnerabilities and compared its performance with other open-source vulnerability scanners. The comparison comprises three metrics: time taken to scan, detection accuracy and consistency. In Abdullah (2020), two open-source web application vulnerability scanners, including Paros and OWASP ZAP, were experimented with for checking vulnerabilities in two vulnerable web applications. The author suggested that the vulnerability assessment tools must constantly be updated to support the discovery of new vulnerabilities that may open up an opportunity for cyber threats. Furthermore, Alsaleh et al. (2017) evaluate the detection performance of two open-source web vulnerability scanners from different perspectives. While the results could

not indicate significant differences between the two scanners, there were differences and inconsistencies between the scanner reports.

Several works on web vulnerability assessment propose tools and techniques to measure vulnerabilities against the 2017 OWASP Top 10 (Amankwah, Chen, Kudjo, & Towey 2020; Khera et al., 2019; Mburano & Si, 2018; Nagpure & Kurkure, 2017; Vibhandik & Bose, 2015). For example, a web vulnerability assessment approach based on a combination of W3AF and Nikto tools was introduced to address security issues in Vibhandik and Bose (2015). The vulnerability assessment performance was measured against the 2017 OWASP Top 10. The authors asserted that combining W3AF and Nikto tools is more effective in detecting vulnerabilities in web applications since a single tool is inadequate to detect different vulnerability types. In addition, the combination can help narrow the scope of security vulnerability detection for complex web applications and servers.

Moreover, Khera et al. (2019) assessed a website's vulnerability in India by applying the 2017 OWASP Top 10 criteria to determine the risk of corporate website threats. Various tools such as Wire shark, Nmap, Metasploit, and Air crack were utilised to assess network security. In the experiment, the attacker can access files within the server by exploiting open ports unrelated to the website's specific operations. Likewise, Nagpure and Kurkure (2017) assessed website vulnerabilities and compared the performance of different tools based on the 2017 OWASP Top 10 criteria, including Burp Suite, OWASP ZAP and Acunetix, according to the tool capabilities in both manual and automation testing methods. High volume and low complexity vulnerability detection and automation testing provide accurate and efficient results. In addition, Amankwah, Chen, Kudjo and Towey (2020) performed a web application vulnerability assessment using open-source software against the 2017 OWASP Top 10 and prioritised vulnerabilities according to severity. The effectiveness of open-source web vulnerability scanners was measured against the 2017 OWASP Top 10 in Mburano and Si (2018). The results were compared further with those from the Web Application Vulnerability Security Evaluation Project (WAVSEP) benchmark.

METHODOLOGY

There are four methodological steps to achieve the research objectives. Figure 1 shows methodological steps in the university's web application vulnerability assessment.

Research Scope Identification

The researchers determined the scope of the university's primary web applications and contacted the person who could authorise the vulnerability assessment. The permission request letter was sent to the university's president, who authorised the university's information technology centre (ITC) director to provide resources as requested for the

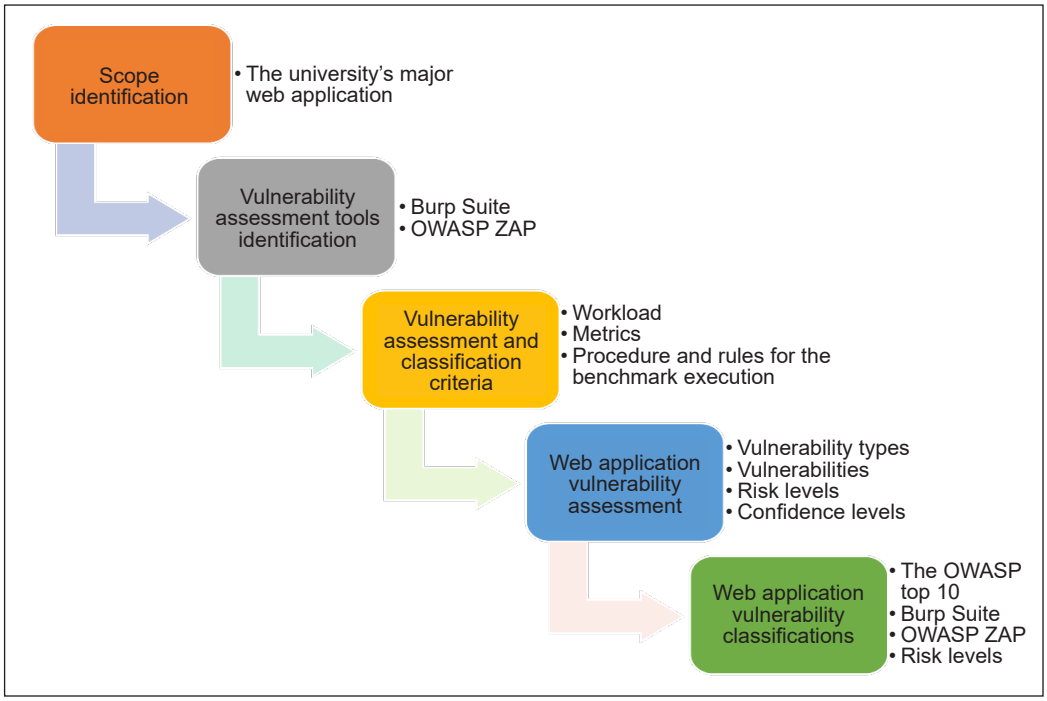

Figure 1. Web application vulnerability assessment methodology

vulnerability assessment. The experiment duration was between January 1, 2022, and February 28, 2022. The experiment was conducted on a system simulated from the university's existing web application to avoid affecting the existing web application and preventing legal offences, according to Thai Netizen Network (2017) and ETDA (2022).

The vulnerability assessment target is one of the university's major web applications storing staff and student personal data. The experiment was conducted on a simulated system to prevent potential issues from the vulnerability assessment in a permitted testing area to search for existing vulnerabilities that may be exposed to cyber threats. The web browser proxy is configured to support the assessment. In addition, boundaries are set within the vulnerability testing program to prevent the program from interfering with other web applications.

Vulnerability Assessment Tools Identification

After identifying the assessment scope, Burp Suite and OWASP ZAP are two vulnerability assessment tools selected for the experiment. The researchers selected these two web vulnerability assessment tools because one is a widely used proprietary software and another is popular open-source software. In addition, these tools similarly classify vulnerabilities into four risk levels: high, medium, low, and informational. The tools also provide confidence levels to support the risk level reliability. While Burp Suite confidence levels

include certain, firm, and tentative, OWASP ZAP confidence levels are high, medium, and low. However, OWASP ZAP also adds 'user confirmed', another confidence level in which users manually confirm vulnerabilities.

Vulnerability Assessment and Classification Criteria

To perform a vulnerability assessment and classification, workload, metrics procedures and rules for the benchmark execution are required as the criteria allow the assessment to be consistent and measurable for identifying and evaluating vulnerabilities (Nunes et al., 2018).

1. Workload is the set of tasks or operations used to evaluate the performance of the system or process, e.g., data processing, database queries, network communication, or other system operations. The university's web application utilises PHP programming language supported by the MySQL database. The application comprises a total of 8,059 files organised within 773 folders. The overall size of the application is 765 MB. However, the quantity of LOC (Lines of Code) for a given file size and number can fluctuate significantly based on several factors (e.g., the programming language, the intricacy of the code, and the use of libraries and data files).
2. Metrics are the measures used to quantify a system's or process's performance, e.g., response time, throughput, CPU utilisation, memory usage, and other performance characteristics. Our vulnerability assessment and classification metrics are the number and percentage of vulnerabilities classified into risk and confidence levels. These metrics can help us prioritise which vulnerabilities need to be addressed and what level of effort should be put into mitigating them.
3. Procedures and rules are the steps for conducting a benchmarking study, e.g., setting up the test environment, configuring the system or process being tested, running the workload, collecting performance data, and analysing the data. Since vulnerability assessment and classification is a process that involves identifying, categorising, and prioritising vulnerabilities, the procedures include both an automatic process, performed by a system such as Burp Suite and OWASP ZAP, and a manual process, performed by a human. In this process, the vulnerabilities reported by the automatic tools are first collected and consolidated. Then, the reported vulnerabilities are manually compared regarding their risk and confidence levels. Please note that vulnerabilities reported by the automatic tools must be triaged and prioritised by a human, helping the organisation focus on the most critical vulnerabilities. When such vulnerabilities identified by both tools must be categorised into the OWASP Top 10, they are manually performed via a repository of vulnerabilities linking to the OWASP Top 10.

Web Application Vulnerability Assessment

Burp Suite and OWASP ZAP were experimented with, and compared their vulnerability assessment performances. This step is critical to test the hypothesis that the university's web application is vulnerable to potential cyber threats and compare the performance of web vulnerability assessment tools. The vulnerability assessment of Burp Suite and OWASP ZAP was conducted utilising a computer system equipped with an Intel(R) Core(TM) i7-10510U CPU, clocked at 2.30 GHz and 16 GB of RAM, running the Windows 10 Pro operating system. The Burpsuite tool typically requires 4 hours to complete a vulnerability assessment, with an average of 10-13 crashes occurring during each assessment. In contrast, the Owasp ZAP tool can complete a vulnerability assessment in approximately 2 hours, with a lower frequency of crashes, averaging 6-8 instances per assessment.

The scanning process of each tool repeats at least three times to ensure that the vulnerability assessment results remain stable. After that, the performance of the two tools in vulnerability assessment will be compared. The reported vulnerabilities are compared at this stage to find similarities and differences. The comparison results at this stage will illustrate how many vulnerabilities are classified into high, medium, and low and what types of vulnerability are discovered.

Web Application Vulnerability Classifications

The vulnerabilities discovered by Burp Suite and OWASP Zap were further classified into the 2021 OWASP Top 10 criteria. First, the vulnerability reports from the two vulnerability assessment tools will be deliberately checked for essential components in the reports. In particular, most web vulnerability reports contain CWE IDs, identifiers referring to vulnerabilities listed by Common Weakness Enumeration (CWE), a widely recognised community listing software and hardware vulnerabilities and their ramifications. For example, CWE-89 means improper neutralisation of special elements used in an SQL Command or 'SQL Injection.' Then, the OWASP Top 10 criteria, their definitions and CWE IDs will also be examined. This step investigates how many university web application vulnerabilities can be categorised in the OWASP Top 10. After that, the classified vulnerabilities between the two tools will be compared.

RESULTS

There are differences in vulnerability assessment performances between Burp Suite and OWASP ZAP.

Risk and Confidence Levels

Burp Suite and OWASP ZAP provide comprehensive reports of vulnerability assessment in which detected vulnerabilities are ranked in risk and confidence metrics. The report,

generated by Burp Suite and depicted in Table 1, presents a university web application's overall vulnerability risk assessment. The assessment revealed 203 vulnerabilities, classified into four categories: high, medium, low, and informational. Additionally, the report indicated three confidence levels in the identified vulnerabilities: certain, firm, and tentative. Five vulnerabilities are at high risk, four at certain confidence and one at firm confidence. In addition, 130 information-level vulnerabilities are at a certain confidence level.

A report was generated utilising OWASP ZAP (Table 2), encompassing 22 vulnerabilities classified into high, medium, low, and informational severity categories. Additionally, the report includes four levels of confidence, specifically user confirmed, high, medium, and low. There is one vulnerability at high risk with medium confidence. In addition, there are 11 vulnerabilities at medium risk with different confidence levels, including two high, seven medium, and two low.

Table 1
Burp Suite vulnerability risk assessment report

		Confidence			
		Certain	**Firm**	**Tentative**	**Total**
Severity	**High**	4 (2.0%)	1 (0.5%)	0 (0.0%)	5 (2.5%)
	Medium	0 (0.0%)	4 (2.0%)	3 (1.5%)	7 (3.5%)
	Low	3 (1.5%)	2 (1.0%)	8 (3.9%)	13 (6.4%)
	Information	130 (64.0%)	39 (19.2%)	9 (4.4%)	178 (87.6%)
	Total	137 (67.5%)	46 (22.7%)	20 (9.8%)	203 (100.0%)

Table 2
OWASP ZAP vulnerability risk assessment report

			Confidence			
		User Confirmed	**High**	**Medium**	**Low**	**Total**
Risk	**High**	0 (0.0%)	0 (0.0%)	1 (4.5%)	0 (0.0%)	1 (4.5%)
	Medium	0 (0.0%)	2 (9.1%)	7 (31.8%)	2 (9.1%)	11 (50.0%)
	Low	0 (0.0%)	1 (4.5%)	3 (13.6%)	1 (4.5%)	5 (22.7%)
	Informational	0 (0.0%)	1 (4.5%)	3 (13.6%)	1 (4.5%)	5 (22.7%)
	Total	0 (0.0%)	4 (18.2%)	14 (63.6%)	4 (18.2%)	22 (100.0%)

Both tools generate vulnerability assessment reports that display risk and confidence levels of vulnerabilities. While Burp Suite's report includes the total of each vulnerability risk level, OWASP ZAP's report presents a summary of risk and confidence levels. Additionally, both tools express identified risk and confidence levels in percentages. In this case, the vulnerability risk and confidence levels reported by Burp Suite and OWASP ZAP are summarised in Table 3.

Table 3
A summary of the vulnerability risk and confidence levels determined by Burp Suite and OWASP ZAP

No.	Risk	Confidence	Burp Suite	(%)	OWASP ZAP	(%)
1	High	High	4	2.0	0	**0.0**
2	High	Medium	1	0.5	1	**4.5**
3	High	Low	0	0.0	0	**0.0**
4	Medium	High	0	0.0	2	**9.1**
5	Medium	Medium	4	2.0	7	**31.9**
6	Medium	Low	3	1.5	2	**9.1**
7	Low	High	3	1.5	1	**4.5**
8	Low	Medium	2	1.0	3	**13.7**
9	Low	Low	8	3.9	1	**4.5**
10	Information	High	130	64.0	1	**4.5**
11	Information	Medium	39	19.2	3	**13.7**
12	Information	Low	9	4.4	1	**4.5**
	Total		**203**	**100.0**	**22**	**100.0**

Vulnerabilities, URL Alerts, and Risk and Confidence Levels

Table 4 shows a report from Burp Suite, which shows the level of threats categorised from High, Medium, Low, and Information levels, along with the number of vulnerabilities discovered. The results also reveal vulnerabilities in different categories.

URLs. As indicated in Table 4, the Burp Suite vulnerability assessment identified 23 distinct vulnerabilities from No.1 to No.23 and generated 203 alerts of URLs. Five vulnerabilities, including No. 1 'SQL Injection' with three alerts and No. 2 'Cleartext Submission of Password' with two alerts, receive high risk and a certain confidence. Three vulnerabilities with 16 alerts are categorised as medium risk, and most confidence is tentative. However, it is essential to note that the vulnerabilities enumerated from No. 12 through No. 23 pertain to informational risks, with a level of confidence ranging from tentative to certain.

A report from OWASP ZAP (Table 5) shows the risk levels categorised into high, medium, low, and informational and the number of vulnerabilities discovered. The results also reveal vulnerabilities in different categories with vulnerable URL alerts. For example, while OWASP ZAP identified 22 vulnerabilities from No.1 to No.22 with 792 URL alerts,

'Cross-Site Scripting' (No.1) is the only vulnerability at high risk with two alerts. There are 11 vulnerabilities at medium risk, with the confidence level from low to high. However, informational risk vulnerabilities receive the highest alerts. In particular, 170 alerts for 'Cookie Slack Detector' (No.18) and 354 for 'User Agent Fuzzer' (No.22).

Table 4
Burp Suite URL alerts report

No.	Burp Suite	Alerts	(%)	Risk	Confidence
1	SQL Injection	3	1.5	High	Certain
2	Cleartext Submission of Password	2	1.0	High	Certain
3	Cross-Site Request Forgery	11	5.4	Medium	Tentative
4	Password Returned in Later Response	1	0.5	Medium	Tentative
5	Session Token in URL	4	2.0	Medium	Firm
6	Vulnerable JavaScript Dependency	7	3.4	Low	Tentative
7	Cookie without HTTP Only Flag Set	1	0.5	Low	Firm
8	Password Field with Autocomplete Enabled	2	1.0	Low	Certain
9	Client-Side HTTP Parameter Pollution (Reflected)	1	0.5	Low	Firm
10	Source Code Disclosure	1	0.5	Low	Tentative
11	Unencrypted Communications	1	0.5	Low	Certain
12	Path-Relative Style Sheet Import	19	9.4	Informational	Firm
13	User Agent-Dependent Response	1	0.5	Informational	Firm
14	Long Redirection Response	6	3.0	Informational	Firm
15	Input Returned in Response (Reflected)	106	52.1	Informational	Certain
16	Cross-Domain Referrer Leakage	4	2.0	Informational	Certain
17	Cross-Domain Script Include	6	3.0	Informational	Certain
18	Frameable Response (Potential Clickjacking)	5	2.5	Informational	Firm
19	HTTP TRACE Method is Enabled	1	0.5	Informational	Certain
20	Backup File	2	1.0	Informational	Certain
21	Email Addresses Disclosed	8	3.9	Informational	Certain
22	Base64-Encoded Data in Parameter	8	3.9	Informational	Firm
23	HTML Does Not Specify Charset	3	1.5	Informational	Certain
	Total	**203**	**100.0**		

Table 5
OWASP ZAP URL alerts report

No.	OWASP ZAP	Alerts	(%)	Risk	Confidence
1	Cross-Site Scripting (Reflected)	2	0.3	High	Medium
2	Absence of Anti-CSRF Tokens	2	0.3	Medium	Low
3	Anti-CSRF Tokens Check	101	12.8	Medium	Medium
4	Application Error Disclosure	1	0.1	Medium	Medium
5	Backup File Disclosure	3	0.4	Medium	Medium

Table 5 *(continue)*

No.	OWASP ZAP	Alerts	(%)	Risk	Confidence
6	Content Security Policy (CSP) Header Not Set	1	0.1	Medium	High
7	HTTP Only Site	1	0.1	Medium	Medium
8	Hidden File Found	1	0.1	Medium	High
9	Insecure HTTP Method - TRACE	77	9.7	Medium	Medium
10	Missing Anti-clickjacking Header	1	0.1	Medium	Medium
11	Parameter Tampering	12	1.5	Medium	Low
12	Relative Path Confusion	32	4.0	Medium	Medium
13	Cross-Domain JavaScript Source File Inclusion	5	0.6	Low	Medium
14	Server Leaks Information via "X-Powered-By" HTTP Response Header Field(s)	1	0.1	Low	Medium
15	Server Leaks Version Information via "Server" HTTP Response Header Field	1	0.1	Low	High
16	Timestamp Disclosure - Unix	1	0.1	Low	Low
17	X-Content-Type-Options Header Missing	1	0.1	Low	Medium
18	Cookie Slack Detector	170	21.5	Informational	Low
19	GET for POST	22	2.8	Informational	High
20	Information Disclosure - Suspicious Comments	2	0.3	Informational	Medium
21	Modern Web Application	1	0.1	Informational	Medium
22	User Agent Fuzzer	354	44.7	Informational	Medium
	Total	**792**	**100.0**		

It can be observed that the number of vulnerability types for URLs generated by Burp Suite and OWASP ZAP is relatively similar, with Burp Suite identifying 1–23 types and OWASP ZAP identifying 1–22 types. However, it should be noted that mapping the vulnerability types between the two tools can be challenging. For example, although Burp Suite categorises 'SQL Injection' and 'Cleartext Submission of Password' as high risk and with a high level of confidence, they are not explicitly included in OWASP ZAP alerts. Besides, some vulnerabilities may be referred to as different issues even though they pertain to the same elements. Additionally, while some vulnerability types may be similarly comprehensible between the two tools, others may differ. For instance, Burp Suite ranks Cross-site request forgery as No. 3 with 11 alerts, while OWASP ZAP ranks Cross Site Scripting (Reflected) as No. 1 with 2 alerts. Furthermore, Burp Suite ranks the Backup file as No. 20 with 2 alerts, while OWASP ZAP ranks the Backup File Disclosure as No. 5 with 3 alerts. Additionally, it is worth noting that the number of URL alerts discovered by both tools is significantly different, with Burp Suite identifying 203 alerts and OWASP ZAP identifying 792 alerts.

In transitioning from a comprehensive vulnerability assessment report to a URL alerts report, a discrepancy regarding the representation of vulnerability numbers has

been identified. Upon comparing vulnerability data, it was observed that Burp Suite identified a consistent number of vulnerabilities across the vulnerability risk assessment (203) and URL alerts (203) reports. However, a discrepancy was identified in the number of vulnerabilities reported by OWASP ZAP in the vulnerability risk assessment (22) and URL alerts (792) reports. Notably, the distribution of vulnerability risk and confidence levels detected by Burp Suite in Table 3 is congruent with the number of URL alerts reported by Burp Suite in Table 4. Conversely, the distribution of vulnerability risk and confidence levels reported by OWASP ZAP in Table 3 is consistent with the number of vulnerability types listed from No.1 to No. 22 in Table 4, suggesting that Burp Suite may have considered vulnerability risk and confidence levels as URL alerts, while OWASP ZAP treated them as distinct vulnerability types.

The 2021 OWASP Top 10 Vulnerabilities

The researchers categorised the vulnerabilities Burp Suite and OWASP ZAP identified according to the 2021 OWASP Top 10 criteria. However, there is a slight difference between the two tools when the detected vulnerabilities must be categorised in the OWASP Top 10. In the Burp Suite vulnerability report, each vulnerability contains an issue background, remediation background, references, and vulnerability classifications. In this case, Burp Suite vulnerability CWE IDs in references will be manually compared with the OWASP Top 10 CWE IDs. If both Burp Suite and the OWASP Top 10 CWE IDs are similar, the Burp Suite vulnerabilities will be classified according to the OWASP Top 10 criteria.

On the other hand, OWASP ZAP provides a descriptive vulnerability report with vulnerability type, source, CWE ID, WASC ID, and reference. Several report components are similar to Burp Suite, containing vulnerability references linking to their definitions and guidelines for reducing risks. However, the significant difference is that the OWASP ZAP report contains sources and their direct links to appropriate criteria of the OWASP Top 10 automatically. In this sense, the authors can understand the criteria and the associated vulnerabilities and effortlessly categorise vulnerabilities according to the OWASP Top 10.The reason might be due to OWASP ZAP being developed by the organisation defining theOWASP Top 10 criteria. Table 6 compares the vulnerabilities reported by the two tools and categorises them by the 2021 OWASP Top 10 criteria.

For example, the number of vulnerabilities from Burp Suite was 189, listed from high to informational risks. There were 5 high risks associated with 2 vulnerabilities, including two for ‘Cryptographic Failures’ and three for ‘Injection’. For ‘Broken Access Control’ considered as the first rank in the OWASP Top 10, Burp Suite identified 15 medium risks, 3 low risks, and 12 informational risks. Please note that there were 1 low risk and 106 informational risks for ‘Injection’ in addition to 3 high risks detected by

Table 6
Vulnerabilities classified into the 2021 OWASP Top 10

No.	OWASP Top 10 (2021)	Burp Suite				Total	(%)	OWASP ZAP				Total	(%)
		H	M	L	I			H	M	L	I		
1	Broken Access Control	0	**15**	**3**	**12**	30	15.9	0	**2**	**2**	**2**	6	1.4
2	Cryptographic Failures	**2**	0	**1**	**8**	11	5.8	0	0	0	0	0	0.0
3	Injection	**3**	0	**1**	**106**	110	58.2	**2**	0	0	0	2	0.5
4	Insecure Design	0	0	0	0	0	0.0	0	**12**	0	**22**	34	7.8
5	Security Misconfiguration	0	0	**1**	**24**	25	13.2	0	**217**	**2**	**170**	389	89.2
6	Vulnerable and Outdated Components	0	0	**7**	0	7	3.7	0	0	0	0	0	0.0
7	Identification and Authentication Failures	0	0	0	0	0	0.0	0	0	0	0	0	0.0
8	Software and Data Integrity Failures	0	0	0	**6**	6	3.2	0	0	**5**	0	5	1.1
9	Security Logging and Monitoring Failures	0	0	0	0	0	0.0	0	0	0	0	0	0.0
10	Server-Side Request Forgery	0	0	0	0	0	0.0	0	0	0	0	0	0.0
	Total	**5**	**15**	**13**	**156**	***189***	**100.0**	**2**	**231**	**9**	**194**	***436***	**100.0**

Burp Suite. Meanwhile, 436 vulnerabilities reported by OWASP ZAP were categorised in the OWASP Top 10. In this case, 'Injection' is the only vulnerability at high risk with two items. In addition, 389 vulnerabilities were classified as 'Security Misconfiguration', of which 231 were medium risks, 9 were low risks, and 194 were informational risks. Burp Suite classifies detected vulnerabilities into six vulnerabilities in the OWASP Top 10, and OWASP ZAP classifies detected vulnerabilities into five vulnerabilities in the OWASP Top 10. As an illustration, Burp Suite identifies 11 vulnerabilities related to 'Cryptographic Failures' classified into two high-risk, one low-risk, and eight informational risk categories. However, this specific vulnerability type is not identified by OWASP ZAP.

Based on the results, certain vulnerabilities identified through Burp Suite and OWASP ZAP can be categorised within the OWASP Top 10. However, it should be noted that there are also instances where both tools detect vulnerabilities, yet they do not fall within the classification of the OWASP Top 10. Those vulnerabilities may include vulnerabilities not considered as severe or widespread as those in the OWASP Top 10. In this case, the OWASP Top 10 is a helpful tool for identifying and prioritising web application security risks, but it is not an exhaustive list of all possible vulnerabilities that tools like Burp Suite and OWASP ZAP can detect.

DISCUSSION

The Number of Informational Risks

Most vulnerabilities not classified in the OWASP Top 10 belong to informational risk but may be worthy of consideration. For example, 'Password Returned in a Later Response' means the web application returns a password in an unencrypted form to the user. 'User-Agent Fuzzer' implies potential bugs in the website code because of response messages to the same URL with a different 'User Agent' header. 'Long Direction Response' means the web application might return a redirection response with 'longer' message content that sometimes contains sensitive information. Moreover, 'Frameable Response (Potential Clickjacking)' should also be considered since this vulnerability might allow the attacker to avoid cross-site request forgery detection, resulting in unauthorised access.

The Reliability of Results

Some limitations might affect the accuracy of vulnerability assessment tools. For example, the number of vulnerabilities assessed against the existing web application may have different results than this study in which the simulated system is placed in front of the firewall to avoid legal issues and impact other network services. Therefore, some types of vulnerabilities might be manipulated and denied entry into the system. These limitations are consistent with Karumbat et al. (2016), who asserted that web vulnerability scanners are not 100% accurate. In addition, there are other potential sources of validity concerns. One example is using web applications with different vulnerabilities, such as those that do not present 'Insecure Design' or 'Security Misconfiguration' issues. In this scenario, the overall number of vulnerabilities identified by Burp Suite may decrease from 436 to 13, while the total number of vulnerabilities identified by OWASP ZAP may decrease from 189 to 164. As a result, it may be challenging to determine which vulnerability assessment tool is more reliable.

The Importance of Medium Risk Vulnerabilities

By comparing the two tools against the OWASP Top 10, Burp Suite could detect more high-risk vulnerabilities than OWASP ZAP. However, on the other hand, OWASP ZAP detected a significantly greater number of vulnerabilities than Burp Suite. Mainly, OWASP ZAP detected medium-risk vulnerabilities more than Burp Suite could several times. Besides, the number of medium risks in OWASP ZAP is more than that of informational risks in OWASP ZAP. The OWASP ZAP medium risk detection is vital, as described by Liu and Wang (2018), who asserted that second-order vulnerabilities are usually ignored but more severe than first-order vulnerabilities. Consequently, the number of medium risk detection should be considered an issue that should be mitigated to protect the university web application from cyber threats.

Inconsistencies

The results also show that some vulnerabilities detected by the two tools are similar and share the same CWE ID but are named differently. In addition, the tools might detect similar vulnerabilities but identify different risk severities. Therefore, using a single tool to assess vulnerabilities may not yield satisfactory results in mitigating cyber threats. Therefore, the benefit of this research is not only the detection and classification of the university web vulnerabilities based on the OWASP Top 10 but also supports some research scholars (Alsaleh et al., 2017; Mburano & Si, 2018), who claim that each vulnerability assessment tool produces different outcomes for the number and severity of vulnerabilities due to detection algorithm differences. Combining different tools is more desirable for detecting more vulnerabilities than a single tool alone, given that web security is paramount.

Vulnerability Risk Mitigation

Vulnerabilities reported from both tools were sent to the responsible IT administrators to assign network security administrators to take action based on the recommendations presented in the report. Therefore, the contributions of this research are not only the classification and comparison of vulnerabilities in the university web application but also the mitigation of risks that cyber threats can exploit.

CONCLUSION

The results confirm that the university web application has vulnerabilities exposed to cyber threats at high and low-risk levels. The vulnerability assessment was experimented with computer-related acts in Thailand, including the Computer Crime Act and Personal Data Protection Act, to avoid any potential effects on the existing university web application. The researchers have measured and compared the performance of web vulnerability assessment tools between Burp Suite and OWASP ZAP in detecting and analysing vulnerabilities in three scenarios, including overall vulnerability risk reports in risk and confidence metrics, vulnerabilities and URL alerts with risk and confidence metrics, vulnerabilities classified into the 2021 OWASP Top 10 vulnerabilities.

Several vulnerabilities were discovered in the university web application. Combining two vulnerability assessment tools, including Burp Suite and OWASP Zap, could detect more vulnerabilities essential for mitigating risks from cyber threats. In this case, Burp Suite and OWASP Zap differ in their vulnerability assessment performance. There are advantages and disadvantages of the tools. The results have revealed differences and inconsistencies in vulnerabilities assessed by both tools. These differences and inconsistencies, however, highlighted advantages and disadvantages helpful for security analysts and administrators to mitigate vulnerability risks to cyber threats. Therefore, combining these two vulnerability assessment tools could detect numerous vulnerabilities with different results and be valuable

for mitigating security risks to the university web application. This research is consistent with Vibhandik and Bose (2015), who conducted various vulnerability assessment tools to test web application vulnerabilities.

ACKNOWLEDGEMENT

The authors express their gratitude to Phuket Rajabhat University, Thailand, for providing support in completing the study.

REFERENCES

Abdullah, H. S. (2020). Evaluation of open source web application vulnerability scanners. *Academic Journal of Nawroz University, 9*(1), 47-52. https://doi.org/10.25007/ajnu.v9n1a532

Alexei, L. A., & Alexei, A. (2021). Cyber security threat analysis in higher education institutions as a result of distance learning. *International Journal of Scientific & Technology Research, 10*(3), 128-133.

Alsaleh, M., Alomar, N., Alshreef, M., Alarifi, A., & Al-Salman, A. M. (2017). Performance-based comparative assessment of open source web vulnerability scanners. *Security and Communication Networks, 2017*, Article 6158107. https://doi.org/10.1155/2017/6158107

Amankwah, R., Chen, J., Kudjo, P. K., & Towey, D. (2020). An empirical comparison of commercial and open-source web vulnerability scanners. *Software - Practice and Experience, 50*(9), 1842-1857. https://doi.org/10.1002/spe.2870

Amankwah, R., Chen, J., Kudjo, P. K., Agyemang, B. K., & Amponsah, A. A. (2020). An automated framework for evaluating open-source web scanner vulnerability severity. *Service Oriented Computing and Applications, 14*, 297-307. https://doi.org/10.1007/s11761-020-00296-9

Darus, M. Y., Omar, M. A., Mohamad, M. F., Seman, Z., & Awang, N. (2020). Web vulnerability assessment tool for content management system. *International Journal of Advanced Trends in Computer Science and Engineering, 9*(1.3), 440-444.

Diogenes, Y., & Ozkaya, E. (2018). *Cybersecurity – Attack and defense strategies: Infrastructure security with Red Team and Blue Team tactics*. Packt Publishing.

Disawal, S., & Suman, U. (2021, March 17-19). *An analysis and classification of vulnerabilities in web-based application development*. [Paper presentation]. 2021 International Conference on Computing for Sustainable Global Development (INDIACom), New Delhi, India.

ETDA. (2022). *Personal data protection act*. Electronic Transactions Development Agency. https://ictlawcenter.etda.or.th/laws/detail/DP-Act-2562

Ibrahim, A. B., & Kant, S. (2018). Penetration testing using SQL injection to recognise the vulnerable point on web pages. *International Journal of Applied Engineering Research, 13*(8), 5935-5942.

Karumba, M. C., Ruhiu, S., & Moturi, C. A. (2016). A hybrid algorithm for detecting web based applications vulnerabilities. *American Journal of Computing Research Repository, 4*(10), 15-20. https://doi.org/10.12691/ajcrr-4-1-3

Khalid, M. N., Farooq, H., Iqbal, M., Alam, M. T., & Rasheed, K. (2019). Predicting web vulnerabilities in web applications based on machine learning. In I. S. Bajwa, F. Kamareddine & A. Costa (Eds.), *Intelligent Technologies and Applications* (pp.473-484). Springer.

Khera, Y., Kumar, D., Sujay., & Garg, N. (2019, February 14-19). *Analysis and impact of vulnerability assessment and penetration testing*. [Paper presentation]. International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon), Faridabad, India.

Liu, M., & Wang, B. (2018). A web second-order vulnerabilities detection method. *IEEE Access, 6*, 70983-70988. https://doi.org/10.1109/ACCESS.2018.2881070

Malekar, V., & Ghode, S. (2020). A review on vulnerability assessment and penetration testing open source tools for web application security. *International Journal of Advanced Research in Science & Technology (IJARST), 2*(3), 30-33.

Mburano, B., & Si, W. (2018, December 18-20). *Evaluation of web vulnerability scanners based on OWASP benchmark*. [Paper presentation]. International Conference on Systems Engineering (ICSEng), Sydney, Australia. https://doi.org/10.1109/ICSENG.2018.8638176

McNab, C. (2016). *Network Security Assessment: Know your Network* (3rd ed.). O'Reilly Media.

Muncaster, P. (2020, September 3). *Northumbria Uni Campus closed after serious cyber-attack*. Information Security Magazine. https://www.infosecurity-magazine.com/news/northumbria-uni-campus-closed/

Muncaster, P. (2021, August 31). *Ransomware may have cost US schools over $6bn in 2020*. Information Security Magazine. https://www.infosecurity-magazine.com/news/ransomware-cost-us-schools-6bn-2020/

Naagas, M. A., Mique Jr, E. L., Palaoag, T. D., & Cruz, J. D. (2018). Defence-through-deception network security model: Securing university campus network from DoS/DdoS attack. *Bulletin of Electrical Engineering and Informatics, 7*(4), 593-600. https://doi.org/10.11591/eei.v7i4.1349

Nagpure, S., & Kurkure, S. (2017, August 17-18). *Vulnerability assessment and penetration testing of web application*. [Paper presentation] International Conference on Computing, Communication, Control and Automation (ICCUBEA), Pune, India. https://doi.org/10.1109/ICCUBEA.2017.8463920

Nunes, P. J., Medeiros, I., Fonseca, J. M., Neves, N. F., Correia, M. P., & Vieira, M. P. (2018). Benchmarking static analysis tools for web security. *IEEE Transactions on Reliability, 67*(3), 1159-1175. https://doi.org/10.1109/TR.2018.2839339

Pavlova, E. (2020). Enhancing the organisational culture related to cyber security during the university digital transformation. *Information & Security, 46*(3), 239-249. https://doi.org/10.11610/isij.4617

Popov, G., Lyon, B. K., & Hollcroft, B. (2016). *Risk Assessment: A Practical Guide to Assessing Operational Risks*. Wiley.

Rahamathullah, U., & Karthikeyan, E. (2021, May 25). *Distributed denial of service attacks prevention, detection and mitigation - A review*. [Paper presentation]. Proceedings of the International Conference on Smart Data Intelligence (ICSMDI 2021), Tamil Nadu, India. http://dx.doi.org/10.2139/ssrn.3852902

Thai Netizen Network. (2017). *Computer crime act 2017 Thai-English Thailand's computer-related crime act 2017 bilingual*. Thai Netizen Network. https://thainetizen.org/docs/cybercrime-act-2017

Ulven, J. B., & Wangen, G. (2021). A systematic review of cybersecurity risks in higher education. *Future Internet, 13*(2), Article 39. https://doi.org/10.3390/fi13020039

Vibhandik, R., & Bose, A. K. (2015, September 21-23). *Vulnerability assessment of web applications - A testing approach*. [Paper presentation] International Conference on e-Technologies and Networks for Development (ICeND), Lodz, Poland. https://doi.org/10.1109/ICeND.2015.7328531

Wear, S. (2018). *Burp Suite Cookbook: Practical Recipes to Help you Master Web Penetration Testing with Burp Suite*. Packt Publishing.