Automated Software Testing and Tool Selection: Case Study Based on Security Testing of Popular E-commerce Applications in Malaysia

M.M.F. Naja[1*], A.R.F. Shafana[2] & A.F. Musfira[3]

[1]*Department of Software Engineering, Universiti Malaya, Malaysia*
[2]*Department of Information and Communication Technologies, Asian Institute of Technology, Thailand*
[3]*School of Multimedia Technology and Communication, Universiti Utara Malaysia, Malaysia*
[1,2,3]*Department of Information and Communication Technology, South Eastern University of Sri Lanka, Sri Lanka*

[1*]mmfnaja@seu.ac.lk, [2]arfshafana@seu.ac.lk, [3]ameermusfi@seu.ac.lk

Abstract- *Advancements in software engineering and software development processes have paved the way for the introduction of new processes and the use of advanced tools for these processes. Automation of software processes and the use of automated tools have gained popularity in the recent past. Thus, the use of automated tools for software testing in ensuring the quality of a software system or application has also found a reach in recent times. Although the importance of these tools has been studied by various researchers in the field, a lack of knowledge on the selection of tools among the practitioners is a challenge. It is because of the availability of an abundant number of tools and the absence of clarity about the tools. Hence this study primarily focuses on investigating the automated tools while trying to identify the better tools for security testing of applications or systems. For this purpose, five e-commerce applications that are popular in Malaysia have been chosen. Also, five tools identified after a thorough study on the tools available for executing automated testing have been used to identify the best tool out of the chosen ones. Although the research work involves studying e-commerce applications available in Malaysia, this research at no instance intends to analyse the applications, rather the tools only.*

Keywords: Software Automation, Test Automation, E-commerce Security, Software Quality, Automated Testing Tools

I. INTRODUCTION

Software Testing is considered as one of the important phases in any software development used to ensure the quality of a software product. In every software development lifecycle (SDLC), the software testing phase is given importance. At the same time, it is said that 50% of the total software development cost is spent on the software testing phase (Gao et al., 2014). While this is being an important note regarding software testing, the emergence of automation practices has paved a way to reduce this cost. Although automation is applied in every phase of SDLC, it has been said that automation in software testing has gained more attention (Hooda and Singh Chhillar, 2015) due to the advantages it thus provides. Apart from that, automation in software testing tends to make the testing process effective while reducing the total cost spent on testing. While talking about software testing, it is mainly about quality assurance.

A number of testing techniques and tools are available which could be used to ensure the quality of a product. But the wise choice in the selection of those testing tools is crucial (Raulamo-Jurvanen, 2017). And when it comes to software like e-commerce and banking, it is mandatory to secure the whole transaction. With that in mind, this study focuses on investigating the importance of security testing, which is one type of testing, and analyze the existing automated tools used in software testing. This research involves performing automated security testing on five well-known e-commerce website widely used in Malaysia for online shopping. After thorough research on automated tools based on security testing, different automated tools have been chosen to test the selected e-commerce sites which affect the ranking of such e-commerce websites. To analyze the existing automated software tools and the importance of selecting the tools, few well-known e-commerce websites, namely, Lazada, Shopee, Lelong, 11Street, and Mudah, have been chosen. Since the websites are all well established, the purpose is not to primarily analyze them but to analyze the tools based on the experience of the websites used, i.e., to check if the analysis of the tools matches the already acquired experience of the users of the websites. In the article, it has been mentioned which website is better for a particular tool, and such information

was only used to analyze the tool itself, not the website.

This research work is intended on analysing the automated testing tools only. This is under no circumstances intended to criticize any e-commerce websites that have been tested using the tools under observation. The opinions expressed are in good faith and while every care has been taken in preparing this article, the authors of this report make no representations and give no warranties of whatever nature in respect of this article, including but not limited to the accuracy or completeness of any information, facts and/or opinions contained therein. The opinions are based on the results obtained only during the testing duration, which may vary with time due to many reasons like change/upgrade of websites used for analyzing the tools, the tools used for testing analysis, etc. sections of this paper.

II. RELATED WORKS AND EXISTING LITERATURE

Although the concept of automated testing and tools used for this have been studied by previous researchers, the realization of which tool could best suit our purpose is still a question. Also, the concept of security testing is vital these days and recent researches on this further proves this. In the work by Pan (2019) on investigating interactive application security testing, he highlights the importance of security testing. Moreover, while we have a look at the work by Alhawi and his co-authors (Alhawi, Akinbi, and Dehghantanha, 2019) on investigating security testing approaches in IoT applications, it highlights that the security testing concept is given importance for a range of applications not only just desktop and web applications only. Considering this importance, the study focus is given to security testing. Apart from this, many existing studies have focused on analyzing e-commerce applications to analyze their usability. Although usability and testing are two different topics, software testing to ensure quality products shall indeed impact the usability of the product. In that way, many existing studies (Hussain et al., 2019; Hussain et al., 2017) have focused on analyzing usability while taking the scope to e-commerce applications. Hence, the scope of this study is chosen to be e-commerce applications. This section of the article discusses few concepts and key terms related to the topic by highlighting the facts discussed in the existing literature.

A. Software Test Automation

The concept of software testing and automation has been a topic of research for a long period. The development in the particular field is yet growing with the introduction of new concepts and practices in the software development industries and software engineering field. Thus, the recent developments in software engineering practices have brought up the concept of automation in almost all the phases of the software development lifecycle (SDLC). As of that the use of automated tools in software engineering and especially in software testing.

In every test activity, it is always essential to find out why an approach is selected. Since software testing is one of the major phases in any software development, it is labor-intensive and expensive. According to the literature, it is stated that testing takes up to 50% of the total cost of any software development. It is sometimes even more than that, according to some literature (Gao et al., 2014). As this is the fact regarding testing cost, it is essential to manage it and that is the main goal of automation testing. Another importance of automation testing, according to literature is, minimizing human error (Jensen, Prasad and Møller, 2013). Mistakes made by human beings become errors that tend to become faults and failures. Another advantage of automated testing is making regression testing easier (Jensen, Prasad and Møller, 2013), meaning that when automation testing is executed to find the errors in any software testing, it makes the process of finding any consequences of any patch works done during a bug fix. Thus, this will ease the problem of overcoming any possible future errors caused by a bug fix. Though these facts highlight the point that automated testing has proven to be a better option for testing, the adoption of automation testing isautomation testing still seems to be challenging due to some prevailing challenges in adopting automated testing tools in software engineering project development.

B. E-commerce and Software Testing

E-commerce is a rapidly growing technology for online shopping and certain online shopping outlets have gained popularity (Hussain et al., 2017). With the exponential increase in such business and the ever-growing demands from customers, the importance of having high high-quality websites also rises; as of this, the importance of enhancing the quality. This is highlighted by Chan et al., as the privacy risk has

a relationship with the purchase intention of the users (Chan et al., 2018). Thus, proper testing of the websites becomes mandatory to ensure their reliable, robust, and high-performing operation. This is the main reason behind the choice of e-commerce application for this case study and since security testing is chosen as the type of testing.

C. *Security Testing*

Security testing is one of the types of software testing executed to ensure that the system or application that is being tested is free of threats or risks or even vulnerabilities that may cause any losses. Security testing is about identifying any possible loopholes and weaknesses of any application that may result in any possible loss of data or confidential information or the information being accessed by any unauthorized party (Mahendra and Muqeem, 2018 The main aim of this testing is meant to find out the threats and investigate the consequences of any potential vulnerabilities that may stop the application from malfunctioning or stop functioning. This also aids in detecting the possible risks and helps to fix them with security features to avoid any negative consequences that may affect the application or system in the future. This is also helpful in implementing all the possible security features to protect the application. Unlike the other testing like the functional test, which is used to prove that a certain function exists and complies with the specification, security testing ensures that certain behavior is not in the application, including reasonably complex testing procedures (Malek et al., 2012). Though, this testing is important to avoid future problems of unauthorized access. As this is the concept behind security testing, to achieve one of the objectives of this study which is to investigate the right tool in testing, security testing is chosen as it is one of the most important types of testing that should always be considered.

III. METHODOLOGY

Intending to investigate the importance of security testing and the selection of automated tools for security testing, this study adopts an approach of performing security testing on certain selected web applications with certain tools selected for testing. This process is performed in a step-by-step manner, starting from tools selection e-commerce application identification to writing test plans up to reporting the results. Key steps in the methodology adopted are selecting the automated tools for testing, identifying e-commerce applications for testing, executing testing, interpreting the results, and comparing them. Further elaborations on the key steps are discussed as follows.

A. Selection of automated Tools for Security Testing

This involves the selection of appropriate tools from a list of tools available. Although there are many tools available for use in online and offline mode providing several facilities to carry out the intended testing, it is hard to choose all and impractical to analyze all of them in practice. The identified tools were screened based on their relevancy of provided features, tool availability as open-source, and recommendation of experts. Figure 01 further depicts the processes involved in this stage.

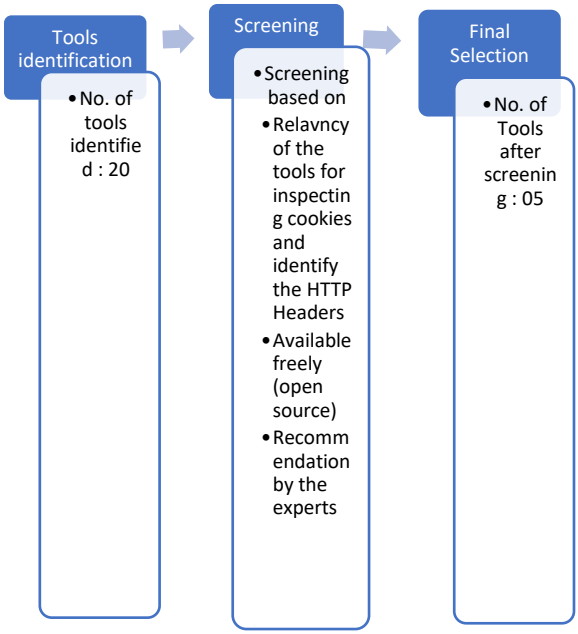

Figure 01: Selection process of automated tools for testing

B. *Selection of E-commerce Application*

For this study, e-commerce applications available in Malaysia are chosen. Though there is a considerable list of applications, only five is chosen as the scope of the study is intended to be with a limited number of applications. Hence, the choice of application is based on the popularity of the application among Malaysian. Details on this process is depicted in figure 02 as follows.

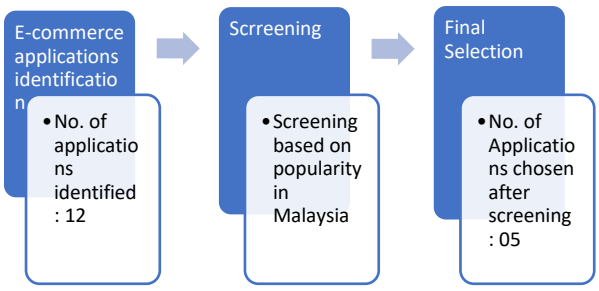

Figure 02: Selection of e-commerce applications for testing

C. *Test Planning and Execution*

At this phase of test planning and execution, the actual testing is performed. The main purpose of creating a test plan is to have a systematic plan logically to test the selected e-commerce websites. Test planning includes the test case preparation and documentation, and the test execution includes performing the test on the selected e-commerce application with the identified test automation tools. Table AA provides a test case description table including all the tests carried out for the study.

Table 01: The sample test case

Test Case ID	Tool_WebApplicationName (Eg: OWASPZAP_Lazada)
Tester	Researchers A, B, and C
Test Purpose	1. To check if the website contains essential security headers. 2. To inspect cookies
Test Procedures	1. Launch the tool. 2. Execute the testing by creating a session for each new session. 3. Interpret the results and analysis. 4. Close each session.
Test Data	URL of the website to be tested: URLs of the selected e-commerce application
Tool Used	OWASP ZAP 2.7.0, Security Header (Online Tool), ImmuniWeb (Online Tool), Sucuri (Online Tool), Pentest tool
Expected Results	To list the missing essential security headers To inspect the cookies
Actual Results	Lists the missing essential security headers and inspects cookies and provides results report for each test case executed
Status	Success

Totally 25 test cases were written. However, a summary of the sample is given in table 01. In the table, the test case ID is denoted with the tool name and the application name. For example, OWASPZAp_ Lazada denotes the test case for testing Lazada's e-commerce application with the OWASPZAP tool. Likewise the rest 24 tests were carried out for the elected five e-commerce applications with selected five tools (05*05=25 Test cases in total).

D. *Results Interpretation and Comparison for Analysis*

After executing the software testing on selected e-commerce applications with the chosen tools, the results produced by the tools were analyzed and the results were compared. Further recommendations and suggestions given in the conclusion section was also based on the results of this step

IV. RESULTS AND DISCUSSION

As discussed in the methodology section of this article, to achieve the objectives of the study, automated software testing was executed on selected e-commerce applications and the results are analyzed in the following sub-sections of this section. Two main security principles have been mainly identified and checked if the tested e-commerce applications comply with the standards and implement the security features.. Although the study focuses on security testing and there are many sub-testing under that, only the availability of HTTP security headers and cookies inspection has been studied during the testing. The main reason for this is that applications selected as the case study for this study are already hosted applications and the primary source code is not available for the public. Therefore, another security feature like penetration testing isn't possible within the scope of the study.

The detailed summary of the testing executed in the selected e-commerce application is discussed as follows. The results are discussed based on the tools used.

A. ImmuniWeb

ImmuniWeb was chosen as one of the tools for security testing for the study. According to the results, this tool is proven to be one of the best tools for security testing as the tool produces a detailed report on the results. This tool provides detail of available HTTP security headers and also inspects the cookies in the application being tested. Apart from that, this tool also produces a summary of the test output and a grade given to the tested application, which implies the level of the application's quality based on the testing it carries out. Detailed reports produced by the tool are tabulated and given in table 02, table 03, and table 04, and figure 03. The detailed test results of all the e-commerce applications selected for this study are given in one shot in the following tables and figures.

Table 02: Summary of available headers in each website listed by ImuniWeb

Security Headers	**Lazada**	**Shopee**	**11street**	**Mudah**	**Lelong**
Strict-Transport-Security	v	v	v	v	v
X-Frame-Options	v	v	v	v	v
X-XSS-Protection	v	v	v	v	v
X-Content-Type-Options	v	v	v	v	v
Expect-CT Feature-Policy	v	v	v	v	v

Table 03: Summary of cookies inspection by ImmuniWeb

Cookie	**Lazada**	**Shopee**	**11street**	**Mudah**	**Lelong**
COOKIE: BROWSERID	-	-	-	-	v
COOKIE:__REQUESTVERIFICATIONTOKEN	-	-	-	-	v
COOKIE: THW	v	-	-		-

Table 04: Summary of test outputs produced by ImmuniWeb

Security Analysis	**Lazada**	**Shopee**	**11street**	**Mudah**	**Lelong**
CMS Security Analysis (No of Issues found)	FAILED	None	None	3	None
GDPR Security Analysis (No of Issues found)	2	1	None	None	1
PCI DSS Security Analysis (No of Issues found)	3	1	None	1	None
HTTP Headers Security Analysis (No of Issues found)	6	6	8	5	5
Content Security Policy Security Analysis (If available)	Missing	Missing	Missing	Missing	Missing
Overall Score	F	C	C	C	C+

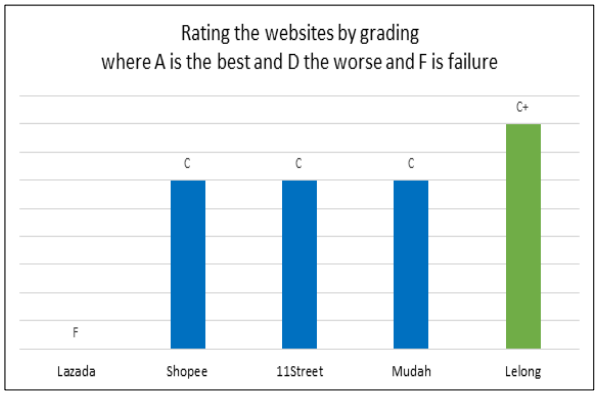

Figure 03: Graph of the analyzed results based on grading by ImmuniWeb

The summary of the test outputs produced by the ImmuniWeb tool is very comprehensive compared to the results produced by other selected tools used here. This is very useful for the developer community to identify the possible doors of penetration and secure them by implementing the security mechanisms that may stop this. Apart from that, the cookies inspection report produced by this tool is also very useful and important when it comes to ensuring the quality and confidentiality of the application users.

B. Security Header

SecurityHeader is yet another tool selected for the study and as implied by the name, this particular tool only produces the list of available HTTP headers and highlights the common HTTP headers which are missing. Apart from this, the tool also rates the application being tested and produces a rating based on grades and this rating seems to be based on the available and missing HTTP headers only. A detailed summary of the results produced by this tool is given in table 05 and figure 04.

Table 05: Summary of the test results produced by Security Header

HTTP Security Headers	**Lazada**	**Shopee**	**11street**	**Mudah**	**Lelong**
Strict Transport Security	V	-	-	V	V
Content-Security-Policy	-	-	-	-	-
X-Frame-Options	-	-	-	-	-
X-XSS-Protection	-	-	-	-	-
X-Content-Type-Options	-	-	-	-	-
Referrer-Policy	-	-	-	-	-
Feature-Policy	-	-	-	-	-
Overall Summary	D	F	F	D	D

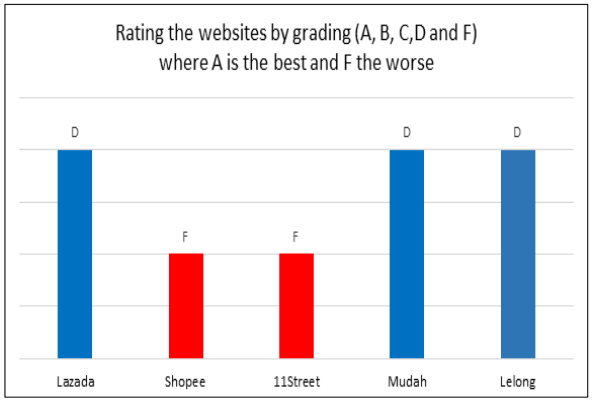

Figure 04: Graph of the analyzed results based on grading by SecurityHeader

C. Sucuri

Sucuri tool also lists the available HTTP headers only and it doesn't even report anything on the important HTTP headers that are missing. Thus, this tool is ought to be less comprehensive compared to all other tools used in the study. The results produced by Sucuri are given in table 06 and figure 05. Figure 05 is the analyzed summary of the rating given by the tool. The rating is based on the risk level analyzed by the tool. Although the detail on how the risk is calculated is not given in the results, it seems it is calculated based on the available HTTP headers as the tool only provides that result.

Table 06: Test results summary by Sucuri

Security Headers	**Lazada**	**Shopee**	**11street**	**Mudah**	**Lelong**
XSS Protection	v	v	v	v	-
Content Type sniffing.	v	v	v	v	-
Strict-Transport-Security security header	-	v	v	v	-
Risk Rate	Low	Low	Low	Mediu	Critical

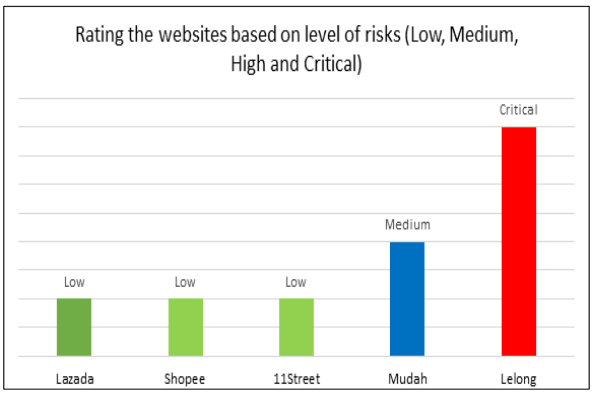

Figure 05: Analyzed results based on level of risk by Sucuri

D. *OWASP ZAP Tool*

The desktop version of the tool was used in the study. Although the tool was chosen for the study based on the recommendation given by the experts in the software testing field, the tool is ought to be complicated compared to the other tools selected for the study. Though, it has a standard user manual to guide the tester. This tool again provides a summary of the available HTTP security headers and inspects the cookies as well. Unlike the other web application tools, this tool does not rate any application and rather provides the test results only The detailed results produced by the tool while executing secrutiy testing are discussed in tables 07 and 08. According to table 8, the tool produces the cookies inspection results of Cookie Without Secure Flag and Cookie No HttpOnly Flag. The tool also gives a detailed description of the important HTTP headers and the solutions for missing headers. Since the tool provides detail of suggestions for missing important HTTP headers, this tool is one of the best tools used in the study. This could be a suggested tool for anyone who performs security testing. This recommendation is based on the experience of the researchers while using this tool for the study.

Table 07: Summary of available headers in each website by OWASP ZAP tool

Security Headers (Included/Missing)	Lazada	Shopee	11street	Mudah	Lelong
X-Frame-Options Header Not Set	V	V	V	-	V
Cross-Domain JavaScript Source File Inclusion	V	V	V	V	-
Web Browser XSS Protection Not Enabled	V	V	V	V	V
X-Content-Type-Options Header Missing	V	V	V	V	V

Table 08: Summary of cookies inspection in each website by OWASP ZAP tool

Cookies	Lazada	Shopee	11street	Mudah	Lelong
Cookie Without Secure Flag	V	-	-	-	-
Cookie No HttpOnly Flag	V	V	-	-	V

E. *PenTest tool*

PenTest tool was another tool selected for the study and this tool, like all other tools chosen for the study, provides detail on the HTPP security headers on the application being tested. The results of that are analyzed and given in Table 09. Apart from this, this tool also gives a detailed description of cookies inspection as analyzed and given in table 10 and a rating for the application being tested. It is given as a graph in figure 06 that plots the rating of all the e-commerce applications used in the study. According to the results, the rating given to the tested applications is primarily based on cookies inspection.

Table 09: Summary of Available Headers in Each Website provided by PenTest Tool

HTTP Security Header	Lazada	Shopee	11street	Mudah	lelong
X-Frame-Options	V	V	V	-	-
X-XSS-Protection	V	V	V	V	V
Strict-Transport-Security	-	V	V	V	V
X-Content-Type-Options	V	V	V	V	V

Table 10: Summary of cookies inspection in each website provided by PenTest tool

Insecure HTTP Cookies	Lazada	Shopee	11street	Mudah	Lelong
__RequestVerificationToken	-	-	-	-	V
BrowserID	-	-	-	-	V
SPC_F	-	V	-	-	-
SPC_T_ID	-	V	-	-	-
REC_T_ID	-	V	-	-	-
SPC_EC	-	V	-	-	-

SPC_U	-	V	-	-	-
SPC_IA	-	V	-	-	-
SPC_T_IV	-	V	-	-	-

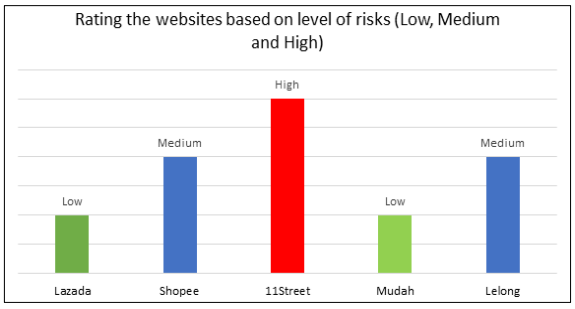

Figure 06: Summary of cookies inspection in each website by Pentest tool

V. CONCLUSION

The results of the security testing carried out using various tools have proven that the selection of automated tools for testing is a tedious task. One of the important facts noted based on the results is which tool is comparatively better based on which criteria. Accordingly, some of the recommendations are derived from the study. It is based on the experience of the researchers while using the tool for the study and results produced by the tools and features supported by the particular tool. Accordingly, the OWSAP ZAP tool wins the race to be the topmost best tool for security testing according to the detailed results produced by the tool. Although the tool is a little complicated to be handled, the technical guidance document of the tool makes this easy. Other than that, the PenTest tool is also considered a wise choice of the tool as it also produces comprehensive cookies inspection results. Since all the tools selected for the study produce a report on the HTTP security headers, the tools are not compared.

Apart from the analysis of the automated security tools, since the study adopts a case study-based investigation and e-commerce applications popular in Malaysia have been chosen for the study, the study results are also useful in identifying the security features implemented in the selected applications. The results of this study hence highlight the quality of the e-commerce applications studied here. Although the main objective of the study is not to rate the applications chosen for the study, the results still pave the way to know about the selected e-commerce applications as well.

Based on investigating automated security testing and proper selection of tools for that, there have been a few realizations by the researchers while carrying out this study and after completing this research. As seen from the results produced by various tools, it is pronounced that one tool is not good enough to test an application/ system. Not much documentation is there to guide through the usage of a particular automated tool or test, which is a trouble for the practitioners. This was based on the experience of the researchers while using the tools chosen for the study. Hence, effective documentation by testers, including their real-life experience of factors affecting a certain tool, for example, can be of great use for future testers.

Apart from this, the results produced by the tools have awakened how well these security features could be implemented to highly secure the applications. The results of the study also highlight the quality of existing e-commerce applications which are popular in Malaysia. Also, this research work paves the way for future research on investigating the suitability of available automated tools for testing software applications or systems for other parameters like performance, functionality, and accessibility.

REFERENCES

Alhawi, O., Akinbi, A. and Dehghantanha, A., 2019. Evaluation and Application of Two Fuzzing Approaches for Security Testing of IoT Applications. *Handbook of Big Data and IoT Security*, pp.301-327.

Chan, S., Ahmad, M., Zaman, I., Omar, S., Ramlan, R. and Tam, C., 2018. Privacy perceptions of online shopping behaviour amongst Malaysian Lazada online shoppers.

Gao, J., Bai, X., Tsai, W. and Uehara, T., 2014. Mobile Application Testing: A Tutorial. *Computer*, 47(2), pp.46-55.

Hooda, I. and Singh Chhillar, R., 2015. Software Test Process, Testing Types and Techniques. *International Journal of Computer Applications*, 111(13), pp.10-14.

Hussain, A., Mkpojiogu, E., Abubakar, H. and Hassan, H., 2019. A Mobile Usability Test Assessment of an Online Shopping Application. Journal of Computational and Theoretical Nanoscience, 16(5), pp.2511-2516.

Hussain, A., Mkpojiogu, E., Jamaludin, N. and Moh, S., 2017. A usability evaluation of Lazada mobile application.

Jensen, C., Prasad, M. and Møller, A., 2013. Automated testing with targeted event sequence generation. *Proceedings of the 2013 International Symposium on Software Testing and Analysis*,.

Mahendra, N. and Muqeem, M., 2018. An Approach for the Inception of Security Testing in the Early Stages of Software Development. *2018 International Conference on Computational and C*Malek, S., Esfahani, N., Kacem, T., Mahmood, R., Mirzaei, N. and Stavrou, A., 2012. A Framework for Automated Security Testing of Android Applications on the Cloud. *2012 IEEE Sixth International Conference on Software Security and Reliability Companion,.haracterization Techniques in Engineering & Sciences (CCTES)*.

Pan, Y., 2019. Interactive Application Security Testing. *2019 International Conference on Smart Grid and Electrical Automation (ICSGEA),*.

Raulamo-Jurvanen, P., 2017. Decision Support for Selecting Tools for Software Test Automation. ACM SIGSOFT Software Engineering Notes, 41(6), pp.1-5.