

E-ISSN: 2582-2160 • Website: <u>www.ijfmr.com</u> • Email: editor@ijfmr.com

Comprehensive Exploration of Web Application Security Testing with Burp Suite Tools

Rhythm Choudhary¹, Jhanvii Rawat², Garima Singh³

¹Assistant Professor, Department of Computer Applications, Maharaja Surajmal Institute, C-4, Janakpuri, New Delhi

^{2,3}Student, Department of Computer Applications, Maharaja Surajmal Institute, C-4, Janakpuri, New Delhi

Abstract

This research paper investigates the pivotal role of Burp Suite tools in web application security testing, aligned with the foundational principles of secure coding practices outlined by the Open Web Application Security Project (OWASP). The paper provides an in-depth overview of Burp Suite tools, emphasizing their features and the significance of web application vulnerability scanning in enforcing robust security practices. The exploration includes a detailed examination of Burp Repeater and Burp Intruder, with a practical demonstration of a Burp Suite cluster bomb attack. This research contributes to advancing the understanding and application of Burp Suite tools for effective web application security testing within the framework of OWASP secure coding practices.

Keywords: Burp Suite, web application security, OWASP, Kali Linux, vulnerabilities, Burp intruder, Burp repeater.

1. INTRODUCTION

In the ever-evolving digital landscape, web application security is paramount due to the increasing complexity of cyber-attacks. This paper underscores the importance of effective testing and vulnerability assessments for ensuring the resilience of web applications. Burp Suite emerges as a versatile penetration testing tool, offering a rich set of features for evaluating the security posture of web services. The research aims to systematically analyze Burp Suite tools, providing insights into their functionalities and their role in security testing. The study employs controlled experiments on representative web applications, leveraging Burp Suite alongside other leading security testing tools.

Overall, this research paper serves as a comprehensive guide for cybersecurity novice and practitioners seeking to leverage the power of Burp Suite tools in effectively testing the security of web applications which contributes to the existing body of knowledge by offering valuable insights, empirical evidence, and critical analysis to enhance the understanding and application of Burp Suite in the realm of web application security.

2. SECURE CODING PRACTICES OWASP

The OWASP Secure Coding Practices Quick Reference Guide offers essential guidelines for developers to write secure code [1]. The goal of the guide, which was created by the Open Web Application Security Project (OWASP), is to assist developers in identifying and address common security



E-ISSN: 2582-2160 • Website: <u>www.ijfmr.com</u> • Email: editor@ijfmr.com

vulnerabilities in their software applications. Focused on preventing security vulnerabilities at the root level, the guide covers various topics, including session management, error handling, output encoding, input validation, access controls, and cryptography. The guide's practicality and concise recommendations make it a valuable resource for developers, helping reduce common vulnerabilities like SQL injection and cross-site scripting. Each topic provides concise and actionable recommendations that developers can easily implement in their code.

Developers can reduce the possibility of common vulnerabilities like SQL injection, cross-site request forgery (CSRF), cross-site scripting (XSS), and unsecured direct object references by adhering to these best practices. It emphasizes input validation techniques, output encoding strategies, secure authentication, and session management practices. It provides recommendations for implementing proper input validation techniques, such as whitelisting, blacklisting, and regular expressions. Output encoding is another critical aspect covered in the guide, which helps prevent XSS attacks by properly encoding user-generated content before displaying it in web pages. The guide provides guidance on using context-specific encoding techniques to prevent malicious script execution. Furthermore, the guide addresses secure authentication and session management, stressing the importance of using strong password hashing algorithms, implementing multi-factor authentication, and securely managing session tokens to prevent unauthorized access. Error handling and logging practices are also emphasized, as proper error handling can help prevent information leakage and assist in identifying and addressing potential security vulnerabilities.

The Quick Reference Guide for OWASP Secure Coding Practices is designed to be concise, easy to understand, and readily applicable. It serves as a valuable resource for developers of all skill levels, providing practical guidance and examples to enhance the security of software applications. Developers can greatly lower the chance of introducing security flaws and help create more robust and secure systems by implementing these secure coding principles [9].

2.1 OWASP Checklist for Secure Coding Practices

The extensive OWASP Secure Coding Practices Checklist is intended to guide developers in implementing secure coding practices to prevent common security vulnerabilities in their software applications. It serves as a practical reference to ensure that applications are developed with security in mind from the ground up [1]. The checklist covers a wide range of security-related areas, including authentication, session management, access control, input validation, output encoding, cryptography, error handling, and more[7]. By adhering to the recommendations outlined in the checklist, developers can significantly reduce the risk of introducing vulnerabilities into their applications.

A. Input validation

Input validation is a critical aspect of secure coding, and the checklist provides guidance on implementing proper input validation techniques [7]. It emphasizes the importance of sanitising and validating data provided by users to stop injection attacks like cross-site scripting (XSS) and SQL injection. The checklist recommends using whitelisting, blacklisting, or regular expressions to validate input and ensure that only expected and safe data is processed by the application. According to OWSAP Secure Coding Practices Checklist [1], for input validation, every data validation procedure must to be carried out on a reliable system, such the server. verification of all information obtained from unreliable sources, such as file streams and databases, encoding of data to a common character set.



B. Output encoding

Output encoding is another important consideration to prevent XSS attacks. The checklist provides recommendations on proper encoding techniques based on the context in which the data is displayed. It emphasizes the need to encode user-generated content to prevent malicious scripts from being executed in users' browsers. According to OWSAP Secure Coding Practices Checklist, follow tried-and-true procedures for every kind of outgoing encoding, clean up any output that uses untrusted data in operating system commands. When it comes to output encoding, it is crucial to perform encoding operations on trusted systems and utilize established routines for different types of encoding. Contextual output encoding should be applied to data originating from untrusted sources, and all characters should be encoded unless confirmed safe. Additionally, output containing untrusted data used in queries and operating system commands should be sanitized appropriately.

C. Secure authentication

Secure authentication and session management are addressed in the checklist to prevent unauthorized access. It recommends the use of strong password hashing algorithms, such as bcrypt or Argon2, and encourages the implementation of multi-factor authentication to enhance security. Proper session management techniques, such as using secure session tokens and enforcing session timeouts, are also highlighted.

D. OWSAP Secure Coding Practices Checklist

The Checklist emphasizes secure authentication and password management. It recommends requiring authentication for all pages, utilizing trusted authentication controls, and enforcing password complexity and length requirements. The checklist also highlights the importance of securely storing passwords, implementing proper password hashing, and securely managing password reset and changing operations. These practices aim to protect user accounts, prevent unauthorized access, and ensure effective password management in applications. According to OWSAP Secure Coding Practices Checklist, authentication should be required for all pages and resources, except those explicitly designated as public, utilize established and tested authentication services whenever possible. If the application manages a credential store, store only cryptographically strong one-way salted hashes of passwords, with the table/file being writable only by the application (avoid using MD5 algorithm if possible). Password hashing should be implemented on a trusted system, such as the server, set short expiration times for temporary passwords and links. These practices ensure secure authentication and effective password management to protect user accounts and prevent unauthorized access.

Access control is essential to ensure that users have appropriate privileges within an application. The checklist emphasizes the importance of implementing role-based access control (RBAC) and least privilege principles to limit users' access to only what is necessary for their intended tasks such as: maintain consistency between server-side implementation and presentation layer rules, encrypt and perform integrity checks on client-stored state data, Limit the number of transactions per user/device to deter automated attacks, periodically re-validate user authorization for long authenticated sessions, assign minimum privileges to service accounts or accounts interacting with external systems. Error handling and logging practices are covered to assist developers in handling errors securely. The checklist emphasizes the importance of providing informative error messages to developers while avoiding



E-ISSN: 2582-2160 • Website: <u>www.ijfmr.com</u> • Email: editor@ijfmr.com

disclosing sensitive information to users. It also encourages the implementation of proper logging mechanisms to facilitate incident response and security analysis. Cryptography is a fundamental aspect of secure coding, and the checklist provides guidance on using cryptographic algorithms and libraries correctly. It emphasizes the importance of using well-established and properly implemented cryptographic functions and storing cryptographic keys securely.

By following the recommendations in the OWASP Secure Coding Practices Checklist, application developers may greatly improve the security posture of their products. It provides actionable guidelines and best practices, allowing developers to incorporate security measures into the software development lifecycle effectively ^[7]. Integrating these secure coding practices from the early stages of development helps to build applications that are resilient against common security threats and vulnerabilities.

3. VULNERABILITY SCANNING TOOL : BURP SUITE

This section explores the utilization of Burp Suite as a web application vulnerability scanning tool, running on the Kali Linux operating system. Kali Linux provides a robust environment for security assessments, aligning with industry best practices. Burp Suite comprises various tools that synergistically contribute to web application security testing. By leveraging the powerful features of Burp Suite within the Kali Linux ecosystem, we were able to effectively identify and analyze vulnerabilities in web applications, aligning with industry best practices for secure coding and testing.

3.1 Burp Suite and its tools

A popular and effective collection of tools for evaluating the security of web applications is called Burp Suite. It is created by PortSwigger, a leading provider of web security solutions. Burp Suite [3] offers a comprehensive range of functionalities that aid in identifying vulnerabilities, analyzing web traffic, and assessing the overall security posture of web applications. The suite includes several tools that work together synergistically to provide a holistic approach to web application security testing. Burp Proxy facilitates traffic interception and manipulation, Burp Spider automates the discovery of application components, Burp Scanner identifies vulnerabilities, and Burp Intruder enables customized attack scenarios. The section provides an overview of Burp Proxy and focuses on Burp Repeater and Burp Intruder, demonstrating their capabilities through practical examples.

Another important tool within Burp Suite is Burp Spider. It performs automated crawling of web applications, discovering and mapping out the various pages and functionality within the target application. By traversing through the application, it helps in identifying potential security vulnerabilities, such as missing input validation or unprotected sensitive information.

Burp Scanner is a highly regarded tool that performs automated security scans on web applications. It applies a wide range of security checks to identify common vulnerabilities, including insecure direct object references, cross-site scripting (XSS), and SQL injection. Burp Scanner combines both active and passive scanning techniques, providing comprehensive coverage in vulnerability detection.

Burp Intruder is a versatile tool that facilitates automated and manual security testing by generating and manipulating custom HTTP requests. By using brute-force attacks, fuzzing, and parameter tampering, users can enable the discovery of vulnerabilities related to input validation, session handling, and more. This tool empowers users to test the application's resilience against various attack vectors.

Burp Suite also includes other useful tools like Burp Repeater, which allows for manual request modification and replay for testing specific scenarios. Burp Decoder aids in decoding and encoding data



E-ISSN: 2582-2160 • Website: <u>www.ijfmr.com</u> • Email: editor@ijfmr.com

to identify vulnerabilities related to input handling and data transformation. Additionally, Burp Comparer assists in comparing responses and identifying differences that may indicate security weaknesses or anomalies.

In order to evaluate the security of web applications, security experts, penetration testers, and developers frequently use Burp Suite, which offers an extensive and integrated set of tools. Its diverse range of features and customizable options enable thorough testing, vulnerability identification, and validation of security controls. By leveraging Burp Suite, organizations can enhance the security posture of their web applications and mitigate potential risks.

3.2 Burp Proxy

One of the primary components is Burp Proxy, which acts as a web proxy server. The Proxy tab in Burp Suite serves as a central hub for monitoring and analyzing web traffic. It captures and analyzes data exchanged between web browsers and servers, providing detailed insights into requests and responses [8]. In Burp Suite's Proxy tab, users can access the primary hub for monitoring and analyzing web traffic in the Community Edition. It provides the ability to observe all the data exchanged between web browsers and the servers of visited sites.

By default, Burp Suite captures traffic from all sites. However, when utilizing the built-in browser provided by Burp Suite, it focuses solely on reporting traffic to that specific browser. This simplifies information management within Burp Suite, as activities in users' regular browsers won't interfere with the generated reports.

The Proxy tab is divided into four sub-tabs: Intercept, HTTP History, WebSocket History, and Options. These sections allow users to capture and analyze traffic for in-depth examination within Burp Suite.

3.3 Burp Repeater

Burp Repeater allows users to intercept and edit specific requests between the client and server, facilitating thorough vulnerability testing. It supports session handling, enabling the testing of authenticated functionality. The server's response is then displayed within the Repeater interface, enabling users to assess the application's behaviour and identify potential security weaknesses The section provides a practical demonstration of Burp Repeater, guiding users through intercepting requests, modifying user interactions, and forwarding requests.

.In addition, Burp Repeater supports session handling, allowing users to maintain the session state while conducting multiple requests. This feature is particularly valuable for testing authenticated functionality and identifying vulnerabilities related to session management.

Burp Repeater offers various features to enhance testing efficiency. Users can save and reload requests, compare different versions of requests and responses, and bookmark important requests for future reference. The tool also supports macro recording and replay, streamlining the process of automating repetitive testing tasks. Furthermore, Burp Repeater supports session handling, allowing users to maintain the session state while performing multiple requests. Burp Repeater also provides a range of features for efficient testing. Users can save and reload requests, compare different versions of requests and responses, and bookmark important requests for later reference. It also supports macro recording and replay, making it easier to automate repetitive testing tasks.

Demo : Burp Repeater



To begin using Burp Repeater, launch the Burp Suite application in Kali Linux and navigate to the Repeater tab. This tab provides an intuitive interface where users can view and manipulate HTTP traffic.

1. Launch Burp's integrated browser



Navigate to the Proxy then Intercept tab within Burp Suite. Toggle the Intercept is off button, changing it to Intercept is on. Enable interception, click the Open Browser option (Launching Burp's integrated browser). This will open Burp's preconfigured browser, designed to seamlessly work with Burp Suite.

2. Intercepting a request

Burp Projec	t Intruder	Repeater	Window He	þ.			2								
Dashboard	Target	Proxy	Intruder	Repeater	Sequencer	Decoder	¢	\rightarrow G	6 http	s://mail.ya	hoo.com/d/			* 1	🛛 😫 E
Intercept	HTTP histo	ory Wei	tSockets history	Options				HOME	VAL	NEWS	FINANCE	000070	ENTERTAINMENT	LIFE	YAHOO PLUS
							-	TURE	WAL	NEIIS	HINANCE	SPORTS	EATERIANMENT	UNE	18/100 PLUS
Forward		Drop	Intercept is	off	Action	Open Browser					_				
							y	ahoo	/mail	Dr	atts × Se	earch in drafts.		Advanced V	Q
								Com	nnca		v	Archive	Nove 1	Delete	🕽 Span 📖
								0.00	,	۰.					
							Ini	box	2		HP In	ndia Ad	Print more, save	e even more	Exclusive offers a
							L	read			me	4	(No Subject) •**	4	
							1	arred							
							50	arreu							I

Using Burp's browser, attempt to visit mail.yahoo.com and observe that the site fails to load. Burp Proxy has intercepted the HTTP request sent by the browser before it reaches the server. Users can find this intercepted request in the Proxy then Intercept tab.

Examining an intercepted request in Burp Proxy, the request is held here, allowing users to analyze and modify it before forwarding it to the target server.

3. Disabling interception

Since browsers often send numerous requests, users may not want to intercept each one. Click the Intercept is on button to switch it to Intercept is off. Proxy Intercept is off. Re-load the page and you will be able to sign into the mail. Within your test email browser try interacting with the user input functions like writing a mail, deleting a mail, making folders etc. All these interaction will be recorded into the target tab's site map.

4. Setting target and scope





Navigate to target and then site map, there are the list of all the activity and navigation performed over the intercepted browser. Now we can set the scope of targeted requests. Click the entry of the choice, for this demo we'll choose the mail.yahoo.com, right click the request and choose "add to scope"

8				Burp S	uite Communi	ity Edition v202	2.9.6 - Tempora	ary Project							ę
Burp Project Intruder	Repeater	Window	Help												
Dashboard Target	Praxy	Intruder	Repeater	Sequencer	Decoder	Comparer	Logger	Extender	Project (options	User opti	ons Learn			
Site map Scope	Issue defin	itions													
iter: Hiding not found ite	ems; hiding (CSS, image a	nd general bina	ry content; hiding	4xx response	ıs; hiding empt	ty folders								1
https://12.ras.yahoo	LCOTT			Host	Met	hod	LIRI.	Params	Status	Length	MME	type	Title		1
https://13.ras.yahoo			ht	tp://mail.vahoo.co	m GET	1			301	527	HTML		t Has Moved		
https://13144298.fl https://16dda3be8c			Ka cafafr:	🗋 http	://mail.yaho	io.com/									
https://26e01cd210					to scope										
https://2ddea74624				Sca											
https://4475b0eb3/ https://61a5b7aff6a					d to intruder		Ctrl+I								
https://6518610.fls	.doubleclick/	net			d to Repeater d to Sequence		Otrl+R								
https://8667f7deb7					d to Sequence d to Compare										
https://86c348b98 https://90737551ee					d to Compare										
https://a.beap.gemi					w response in										
A														0.0	ċ
				Burp S	uite Communi	ty Edition v202	2.9.6 - Tempora	ary Project						-00	
urp Project Intruder		Window													
Dashboard Target	Proxy	Intruder	Repeater	Sequencer	Decoder	Comparer	Logger	Extender	Project o	options	User opti	ons Learn			
ite map Scope	Issue defin	itions													
Target Scope															
Paste URL Load				Burg S	uite Communi	ty Edition v202	2.9.6 - Tempera	ury Project							
rp Project Intruder	Repeater	Window	Help	out a		.,	E200 - Tempere								1
Dashboard Target	Praxy	Intruder	Repeater	Sequencer	Decoder	Comparer	Logger	Extender	Project o	ptions	User optic	ons Learn			
Site map Scope	Issue defin	itions				ope Proxy traff		Be easily							
iter: Hiding out of scope	and not four	ulitame: hid	ina CSS image						_						1
https://mail.yahoo.				Host	Met		URL	Params	Status ^	Length	MIME	type	Title		15
> 🥹 / > 🧮 td_api				tps://mail.yahoo.c		1			200	15278	HTML	Yahoo Ma	i.		1
> 🧮 apps			ht	tps://mail.yahoo.c tps://mail.yahoo.c	om GET	/digemin	ectly?brandUrl Lapi/?adCount	= √	200 200	845 70522	JSON JSON				
> 📒 comscore_beao > 🥹 d	n		h	tps://mail.yahoo.c tps://mail.yahoo.c	om GET om GET	/d/gemin	i_api/?adCount i_api/?adCount	= V = V	200 200	17702 63499	JSON JSON				
> 🛑 d > 🧮 psearch			ht	tps://mail.yahoo.c	om GET	/d/gemin	i_api/?adCount-	= √	200	63975	JSON				
> 🧮 public			ht.	tps://mail.yahoo.c tps://mail.yahoo.c	om GET om GET	/d/gemin /d/gemin	i_api/?adCount i_api/?adCount	- 1	200	63947 63968	JSON JSON				
pr.gif b tdv2_fp			ht	tps://mail.yahoo.c	om GET	/d/gemin	i_api/?adCount i_api/?adCount	=	200	70947 70481	JSON JSON				
> ws > xobni				tps://mail.yahoo.c			i_api/?adCount i_api/?adCount		200	10810	15ON				
/ 🔤 xaon										in the	•				
					ponse							Inspector	• • I	÷⊚	2
				Pretty Raw	Hex					6	in ≣	Request Attribu	tes	2	~
				Host: mail.y	ahoo.com							Request Header	5	16	v
			1	Upgrade-Inse User-Agent:	Mozilla/5	.0 (Windows	NT 10.0; 1	win64; x64) Applew	ebKit/53	7.35	Response Head	HTS.	18	~
				(KHTML, like Accept:	e Gecko) C	hrome/107.0	0.5304.107	Safari/537	.36			maporse ricadi		10	1
				text/html,ap bp,image/apr	plication	/xhtml+xml, .8.applicat	application	n/xml;q=0.	9,image// v=b3:g=0	avif,ima .9	ge/ve				
				Sec - Fetch - Si	te: none		er of ar Augus	area and a first of a							
				Sec-Fetch-Mc		ard				0.0	atches				
				98 E C	search.					01	res LINES				

Now you can filter the request which only involves yahoo mail.

5. Navigating the user interaction

🔲 🗖 🎝 🍏 🕅 🗸 🗍 2 3 4						4.I.					
	Burp Suite C	ammunity b	dition v2022.9.6 - Temporary	Project							
rp Project Mouder Repeater Window Help Isobboard Target Provy Intruder Repea	er Sequencer De	coder (Comparer Looper I	Nester	Project	and here	User options	Learn			
ternap Scope Issue definitions	ai seferirri ol	courr v	roublia rolla i	LALOSOFT	Pageri	opsure	osa strosie	Learn			
emap scope asserterindons											
			Pray traffic a diables	Re-enable							
er: Hiding out of scope and not found items; hiding CSS, im	age and general binary con	tert; hiding	Accresponses; hiding empty	folders							
https://mail.yshoo.com	Host	Method	URL	Params	Status	Length	MME type		Title		
@ 1	https://mail.yahoo.com	GET	1		200	15278	HTML	Yahoe Mail			
tdapi	https://mail.yahoo.com	GET	/apps/directly/brandUrl+_		200	845	JSON				
comscore beacon	https://mail.yahoo.com	GET	idigenini_api?tsKcunt+.		200	10726	/50N				
Ø #	https://mail.yahoo.com	GET	(digenini_api?adCount+.		200	63908	/50N				
	https://mail.yahoo.com	GET	Idiperrini_api?adCount=.		500	63930	250N				
psearch	https://mail.yahoo.com https://mail.yahoo.com	GET	Illiperrini, api7sdCount= Illiperrini, api7sdCount=		200	70522 63854	250N 250N				
a reblic	https://mail.yahoo.com	GET	Idomini aprild.turt		200	17702	250N				
D page	https://mail.sahoo.com	GET	Illipmini api/adcourt=.		200	63499	250N				
10/2 fp	https://mail.yahoo.com	GET	(diperrini_api?adCount=_		200	63975	JSON JSON				
to ws	https://mail.yahoo.com	GET	iditertiti api?adCourt=_		200	63651	250N				
V 🗖 13	substitues builderen	96.1	indaunit designed and a								
- 😌 batch						(and the	-				
Hannahary1Y0NjrRajGr	Request Response	e.					i 🖬 🖬	pector		÷ @	
WebKtFormBoundary2Kd66c5Erk WebKtFormBoundary41fk/20R3a	Protty Ram H					8	vi =	auest Amribate			
WebKitFormBoundary4m6IVL58a	1 GET / HTTP/2					-		Park Himpere	·	2,	
WebKitFormBoundary5Ae89TWA	2 Host: nail.vaho	6 . COM						pest Headers		1963	
WebKitFormBoundarySBMIapYaid	- 3 Upgrade-Insecur		a: 1				100	STERN HERDERS		1382	
WebKitFormBoundaryBT7y#U0.02;	4 User Agent: Moz	illa/5.0	(windows NT 10.0; wi	n64; x64	Applex	Abkit/S3	7.36	ponse Header	2	18	
WebKitFormBoundaryEEFBmAfos		ckal Chro	me/107.0.5304.107 Sa	fari/537	,36			portie reason	· · · ·	100	
😸 WebKitFormBoundaryF5Hr7h2A2v	5 Accept:										
WebKilformBoundaryGV2t5i8Fm ³			tml+xal,application/				ige/wa				
😸WebKitFormBoundaryH7ApriWith1	<pre>cp,lsage/aprg.* E Sec-Fetch-Sate:</pre>		application/signed-e	xchange;	And 22 dec	.9					
WebKtFormBoundarytvD4NDe8/F	7 Sec-Fetch-Mode:										
WebKitFormBoundaryKUvSYOaTlo WebKitFormBoundaryKUvSYOaTlo											
	(?) (6) + + 5a						natches				

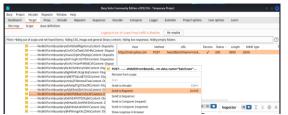
After choosing any of the request navigate to "ws" which has the another folder "v3" within that the "batch" is the folder that has all the user interaction records.



E-ISSN: 2582-2160 • Website: www.ijfmr.com

• Email: editor@ijfmr.com

6. Forwarding the request



Right click on the request and choose "send to repeater"

😫 🔲 💼 🍃 🍪 🍽 v 📘 2 3 4 🛛 🖪 🧞 🗉 🖷	يسامله بفي م	🍽 🔌 🌲 🖀 3:34 🗎 🖨 🚱
🛐 Burp Su	te Community Edition v2022.9.6 - Temporary Project	
Burp Project Intruder Repeater Window Help		
Dashboard Target Proxy Intruder Repeater Sequencer	Decoder Comparer Logger Extender Project options User opti	
1 × 2 × 3 × 4 × +		Q. I
Send Cancel < * > *	Target: h	ttps://mail.yahoo.com 🖉 HTTP/2 💮
Request	Response III = =	Inspector 🔳 🗉 🗄 🛞 🗡
Pretty Raw Hax	Pretty Raw Hex Render 🗊 🖬 🖩	Selection 9 A
d=AQABBFSFmmQCEPzukMBqJ7HDvZRSyRkfR0oFEgEACALSn2TIZNv6QD IB eMBAAcIVTMaZBkfR0oIDxXz0KBm7D00tM0o0CPYYvkBBvoB0A6S=A		Selected text
QAAAtbHjeCUIGDrOSPGYHVY33v6j=WCPLD; cmp= t=168794480536j=C6u=1YNN; gpp=CBABBg=EVoIgAC4.QA; gpp_mid =8; gam.id=		newfalder
y - SMTeMPNO2LI QAI 6K2LNuk APS202LX3Y aPk awDWB10xAAd 0kNawA ; tbla_id=		Decoded from: Select V 🕘
Oclass77:07fb-424e-ba92-8ad54e0f406d-tuctb940d2e; Beekhov35i9llak6b=45du=hn4ySttYFHM_ekogH056s=mo6i=ErPQoCbs NAa155hA19h; CUE=A0EACA1km31kyErvgH5; YH.CLIENT_PARAME		Cancel Apply changes
ih34A78h22ronth22h341520h20h42h341280h20h22h341		Request Attributes 2 ~
68785022228970 4 Content-Length: 441		Request Query Parameters (5) ~
5 Sec-Ch-Ua: "Chronium";v="107", "Not=A7Brand";v="24" 6 Accept: application/json		Request Body Parameters 1 ~
7 Content-Type: multipart/form-data; boundary=WebKitFormBoundary2Kd66c5Erk480zPz		Request Cookies 17 v
<pre>0 Sec-Ch-LA-Mobile: 70 9 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) ApplewebKit/537.36 (NOTML, like Secko)</pre>		Request Headers 35 ~
Chrome/107.0.5304.107 Safari/537.36 10 Sec-Ch-Ua-Platform: "Linux"		
⑦ ⑧ € → Search 0 matches	⑦ ◎ € → Search. 0 matches	
Ready		

7. Modifying the user request

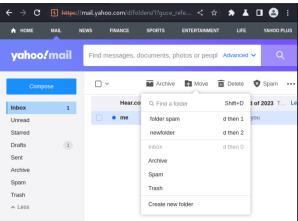




Within this request we can find that the user created a new folder and named it "new folder" we can change the name of this folder to "folder spam" which will create a new folder with the name mentioned. Click send and we can see that the response of our modified request is shown on the response section.

Burp Suite C	mmunity Edition v2022.9.6 - Temporary Project	
1× 2× 3× <u>4×</u> +	ader Comparer Lagger Extender Project options User options Learn	Q
Send (Cancel < + >+	Tarpet: https://mail.yahea.com 🖉	нттр/2 ÷ ©
Pretty Raw Hex 🔂 Vi =	Pretty Raw Hex Render Vi Request Attributes	2
S Sec-Ch-Ua-Mobile: 70 9 User-Agent: Mozilla/S.O (Windows NT 10.0; Win64; x64) ApplewebSit/537.36 (WHML, like Gecko) Chrome/107.0.5304.107	2 Content-Type: application/json 3 X-Server-Response-Time: 290ms Request Query Parameters	5
Safari/537.36 0 Sec-Ch-Ua-Platform: "Linux" 1 Origin: https://mail.vahos.com	4 X-Proxy-Server-Response-Time: SBms 5 X-Server-Chain: 1ws20021.mail.mel.vahoo.com,jws600015.mail.sc3.vah	1
l Drigin: https://wall.yahos.com 2 Sec.Fetch.Site: same-origin 3 Sec.Fetch.Mode: cors	00.com Request Cookies 6 X-Server-Timestamp: Tue, 27 Jun 2023 07:39:06 OMT	17
Sec-Fetch-Dest: empty Referer: https://mail.yahop.com/	7 Kcomm_tags: force.farm=3206,Zm9yr23JJZmPybT02MjA2 8 Vary: Accept-Encoding 9 Content-Length: 918	35
Accept-Encoding: gzip, deflate Accept-Language: en-US,en;q=0.9	9 Content-Length: 918 10 X-Xss-Protection: 1; mode=block Response Headers 11 X-Content-Type: Options: nosniff	16
	12 Reference Holips, in or reference when dowgrade 13 Reference - 10 and 20 07 3500 00 10 Strict Transport Security; nar-apa-01530000 10 Strict 13 D 10 Forward 15 D 10 For	
⑦ ⊕ + Search. 0 matcher	⑦ ⊕ ← → Search. 0 matches	

Go to the browser to see the actual folder that we created by modifying intercepted request from Burp Suite.



3.4 Burp Intruder

Burp Intruder [5] is a powerful tool designed for automated and manual security testing, supporting various attack types such as brute-force attacks and fuzzing. The section introduces the capabilities of Burp Intruder, emphasizing its versatility in generating and manipulating custom HTTP requests for vulnerability discovery. It is designed to automate and streamline the process of performing various types of attacks on web applications, with the goal of identifying vulnerabilities and weaknesses. Intruder allows security testers to customize and automate attacks by defining payloads, specifying attack positions, and configuring various attack options. It supports multiple attack types, including Sniper, Battering Ram, Pitchfork, and Cluster Bomb, each catering to different attack scenarios. With Intruder, testers can manipulate parameters, headers, and other request components to launch attacks like brute forcing, fuzzing, and parameter manipulation. It enables testers to iterate through a list of payloads, sending multiple requests to the target application and capturing the responses for analysis.

The tool provides extensive options for payload customization, including payload positions, payload lists, payload processing rules, and payload encoding. It also offers response handling options for filtering, marking, and extracting interesting data from the responses.

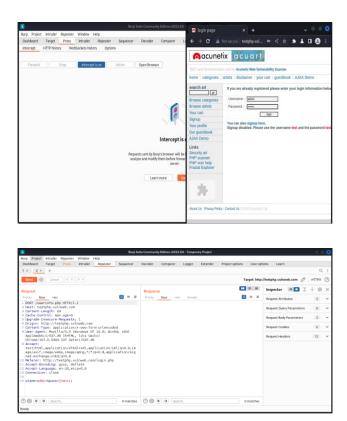
Intruder offers a flexible and intuitive user interface that allows testers to easily configure and launch attacks, monitor the progress, and analyze the results. It provides detailed insights into the success or failure of each attack attempt, enabling testers to identify potential vulnerabilities or areas of concern within the target application. Using Burp Suite Intruder, security testers can efficiently test the security



of web applications, uncover potential vulnerabilities, and help organizations improve their overall security posture.

1. Demo : cluster bomb attack

The Burp Suite Intruder tool offers a powerful capability called Cluster Bomb attack [10], which allows for performing a targeted and extensive attack on a specific parameter or input field of a web application. A Cluster Bomb attack is a form of brute force attack that involves generating multiple payloads and iterating through them systematically. The demo guides users through the steps of preparing the request, defining payloads, configuring Intruder settings, launching the attack, and analyzing results.



1. Prepare the Request: Start by capturing the target request in Burp Suite Proxy or import it manually. Identify the parameter users want to attack and configure it to be the Intruder target.

Burp Project Intruder Repeater Window Help Dashboard Target <u>Proxy</u> Intruder Repeat		Logger Extender Project option	ns User options Learn		
Intercept HTTP history WebSockets history O	ptions				
Request to http://testphp.su/nweb.com.80 [44.228.249. Forward Drop Intercept is on	3] Action Open Browser		•	HTTP/1	0
Pretty Raw Hex 1 POST /userinfo.php HTTP/1.1		🗐 w 🗉	Inspector	÷ (6)	\times
2 Host: testphp.vulnweb.com 3 Content-Length: 24			Request Attributes	2	~
4 Cache-Control: max-age=0 5 Upgrade-Insecure-Requests: 1			Request Query Parameters	0	×
6 Origin: http://testphp.vulnweb.com 7 Content-Type: application/x-www-form-urler 8 User-Agent: Mozilla/5.0 (Windows NT 10.0;		like (aska)	Request Body Parameters	2	×
Chrome/107.0.5304.107 Safari/537.36 9 Accept:	ALTON, AND Approximately and an inc.	(The decko)	Request Cookies	0	~
<pre>text/html,application/xhtml+xml,applicatio /signed-exchange;v=b3;g=0.9</pre>		ge/apng,*/*;q=0.8,application	Request Headers	12	~
<pre>10 Referer: http://testphp.vulnweb.com/login. 11 Accept-Encoding: gzip, deflate</pre>	, php				
12 Accept-Language: en-US,en;q=0.9 13 Connection: close					
14 15 <mark>unane=admin&pass=jhanvii</mark>					



E-ISSN: 2582-2160 • Website: <u>www.ijfmr.com</u> • Email: editor@ijfmr.com

ihboard Targ	et Praxy	Intruder	Repeater	Sequencer	Decoder	Comparer	Logger	Extender	Project options	User options	Learn	
2 × +												
itions Payloa	ds Resource	Pool	Options									
Payload Sets												Start att
You can define o	ne or more paylo	ad sets. Th	e number of pay	load sets depen	ds on the atta	ck type defined i	in the Position	ns tab. Various	payload types are a	vailable for each pa	ryload set, an	d each payload type
be customized in	n different ways.											
Payload set:			Payload cour	t: 7								
Payload type:		_	Request court	± 42								
	•											
Payload Opti	ons [Simple lis	t]										
This payload typ	e lets you configu	re a simple	e list of strings th	nat are used as	oayloads.							
Paste	admin											
	user1											
Load	user2 aadmin											
Remove	newser											
Clear	user3											
Deduplicate	test											
Add	Enter a new i	4.00										
1.000												
Payload Proc You can define r	ules to perform w					ty Edition v2022	2.9.5 - Tempo	rary Project				
Payload Proc	ules to perform v ter Repeater		essing tasks on Help Repeater			ky Edition v2022 Comparer	2.9.5 - Tempo Logger	rary Project Extender	Project options	User options	Learn	
Payload Proc You can define r Project Intrue	ules to perform v ter Repeater	Window	Help	Burp S	iuite Communi				Project options	User options	Learn	
Payload Proc You can define r Project Intrue hboard Targe	ules to perform v fer Repeater et Proxy	Window Intruder	Help	Burp S	iuite Communi				Project options	User options	Learn	
Payload Proc You can define r Project Intrue hoord Targe 2 × + Rions Payloa	ules to perform v fer Repeater et Proxy	Window Intruder	Help Repeater	Burp S	iuite Communi				Project options	User options	Learn	Start att
Payload Proc You can define r Project Intruc hooard Targe 2 × + tions Payloa Payload Sets You can define o	ules to perform v fer Repeater et Prœγ ds Resource ne or more paγio	Window Intruder Pool	Help Repeater Options	Burp Sequencer	litte Commun Decoder	Comparer	Logger	Extender				Slart att
Payload Proc You can define r Project Intruc hboord Targe 2 × + Rions Payload Sets	ules to perform v fer Repeater et Prœγ ds Resource ne or more paγio	Window Intruder Pool	Help Repeater Options	Burp Sequencer	litte Commun Decoder	Comparer	Logger	Extender				
Payload Proc You can define r Project Intruc hooard Targe 2 × + tions Payloa Payload Sets You can define o	les to perform v fer Repeater tt Proxy ds Resource ne or more paylo different ways.	Window Intruder Pool	Help Repeater Options	Burp Sequencer	litte Commun Decoder	Comparer	Logger	Extender				
Payload Proc You can define r Project Intruc hboard Targe 2 × + tions Payloa Payload Sets You can define o be customized in	ler Repeater tt Proxy ds Resource ne or more paylo different ways.	Window Intruder Pool ed sets. Th	Help Repeater Options e number of payl	Burp 5 Sequencer load sets depen t: 6	litte Commun Decoder	Comparer	Logger	Extender				
Project Intrue Project Intrue	ler Repeater tt Proxy ds Resource ne or more paylo different ways.	Window Intruder Pool ad sets. The	Help Repeater Options e number of payl Payload court	Burp 5 Sequencer load sets depen t: 6	litte Commun Decoder	Comparer	Logger	Extender				
Project Intrue Project Intrue	ler Repeater et Proxy ds Resource ne or more paylo different ways.	Window Intruder Pool ad sets. The	Help Repeater Options e number of payl Payload court	Burp 5 Sequencer load sets depen t: 6	litte Commun Decoder	Comparer	Logger	Extender				
Project Intruc Vou can define r Project Intruc Debaad Targe Payload Sets You can define o be customized in Payload Sets 1 Payload type: 1	Ser Repeater et Proxy ds Resource ne or more payloid different ways.	Window Intruder Pool ed sets. The V	Help Repeater Options e number of payl Payload cour Request cour	Burp S Sequencer load sets depen t: 6 t: 42	Ute Commun Decoder	Comparer	Logger	Extender				
Payload Proc You can define r Project Intrus 2 x + Rations Payload Targe Payload Sets: Payload sets: Payload sets: Payload sets: Payload sets: Payload type:	ter Repeater t Pray ds Resource ds Resource drifferent ways. i imple list income law of the list income list income law of the list income l	Window Intruder Pool ed sets. The V	Help Repeater Options e number of payl Payload cour Request cour	Burp S Sequencer load sets depen t: 6 t: 42	Ute Commun Decoder	Comparer	Logger	Extender				
Payload Proc. You can define r Project Intrus 2 x + tions Payload Sets You can define r be customized in Payload Sets Payload set: 2 Payload optis This payload optis This payload typ Paste	ter Repeater t Proxy ds Resource ne or more paylou idfferent ways. t mple list n lets you configu admin	Window Intruder Pool ed sets. The V	Help Repeater Options e number of payl Payload cour Request cour	Burp S Sequencer load sets depen t: 6 t: 42	Ute Commun Decoder	Comparer	Logger	Extender				
Payload Proc You can define r Project Interue 2 + + Xou can define r Payload Sets Payload Sets Payload Sets Payload Optio This payload typ Paste Load	ter Repeater ter Proxy	Window Intruder Pool ed sets. The V	Help Repeater Options e number of payl Payload cour Request cour	Burp S Sequencer load sets depen t: 6 t: 42	Ute Commun Decoder	Comparer	Logger	Extender				
Payload Proc. You can define r Project Intrus Payload Sets Payload Sets Payload set: Payload optin This payload typ Paste	ler Repeater tt Proxy ds Resource ds Resource ds Resource imple list left syou configue passance passance passance	Window Intruder Pool ed sets. The V	Help Repeater Options e number of payl Payload cour Request cour	Burp S Sequencer load sets depen t: 6 t: 42	Ute Commun Decoder	Comparer	Logger	Extender				
Payload Proc You can define r Project Interue 2 + + Xou can define r Payload Sets Payload Sets Payload Sets Payload Optio This payload typ Paste Load	ter Repeater ter Proxy	Window Intruder Pool ed sets. The V	Help Repeater Options e number of payl Payload cour Request cour	Burp S Sequencer load sets depen t: 6 t: 42	Ute Commun Decoder	Comparer	Logger	Extender				
Payload Proc You can define r Project Intru- hoord Tarry Payload Sets Payload Sets Payload Sets Payload set: 2 Payload set: 2 Payload sets Payload s	ter Repeater ter Proxy	Window Intruder Pool ed sets. The V	Help Repeater Options e number of payl Payload cour Request cour	Burp S Sequencer load sets depen t: 6 t: 42	Ute Commun Decoder	Comparer	Logger	Extender				
Payload Proc You can define r Project Intruo Design of the second second Payload Sets Payload Sets Payload Sets Payload Sets Payload Sets This payload type Design of the second second second Payload Sets Payload Sets Payload Sets Payload Sets Payload Sets Payload Sets Payload Sets Catal Remove Clear	ter Repeater ter Proxy	Window Intruder Pool ad sets. Th v v v	Help Repeater Options e number of payl Payload cour Request cour	Burp S Sequencer load sets depen t: 6 t: 42	Ute Commun Decoder	Comparer	Logger	Extender				
Peyload Proc You can define r Project Intruc bloosd Targy 2 * + Kions Payload Sets You can define a bloosd Cates Payload Sets Payload	ler Repeater tt Prory ds Resource ds Resource ads Resource imple list imple list imple list e lets you configu padmin password powd 22 pod test	Window Intruder Pool ad sets. Th v v v t] t] t t t t t t t t t t t t t t	Help Repeater Options e number of payl Payload cour Request cour	Burp S Sequencer load sets depen t: 6 t: 42	Ute Commun Decoder	Comparer	Logger	Extender				

2. Define the Payloads: Create a list of payloads that will be used for the attack. These can include common usernames, passwords, or any other relevant data that users want to test. Ensure the payload list covers a wide range of possibilities.

5						Burp	Suite Communi	ity Edition v2023	2.9.6 - Tempo	rary Project					
Burp	Project	Intruder	Repeater	Window	Help										
Des	hboard	Target	Proxy	Intruder	Repeater	Sequencer	Decoder	Comparer	Logger	Extender	Project options	User options	Learn		
1 ×	2 ×	+													Q. 1
Pos	tions	Payloads	Resour	ce Pool	Options										
0	Choose	an attacl	type												Start attack
	Attack typ	e: Cluste	r bomb												
0	3 Cont 4 Cach 5 Upgr 6 Orig 7 Cont 8 User 9 Acce text 10 Parts 11 Acce 12 Acce 13 Cont 14	Pr th Batter This us PIG Plechf This at at at controls at a	tack uses a ing ram uses a single ork tack uses n aneously, s r bomb tack uses n permutat Mozilla pplicati tp://tes ding: gz uage: en close	set of paylo nultiple paylo nultiple paylo nultiple paylo /5.0 (Wir on/xhtnl+	ads. It iterates ti coad sets. There i first payload fro and sets. There i and combination idours NF 10. writ, applica web.com/log ite d.9	rough the paylo is a different pay in each set, ther is a different pay i are tested. 0; Win/54; x0 tion/xml;q=0	ads, and place load set for ea the second p load set for ea (4) App Lekie	s the same pay ch defined posit iyload from eac ch defined posit ckit/537.36	ion (up to a n h set, and so ion (up to a n (KHTML,	of the defined p naximum of 20 on. naximum of 20 Lake Gecko)	then each payfoad i ayload positions at o 0. The attackiterate 0. The attackiterate 0. The attackiterate 0. The attackiterate 0. On one /107.0.	nce. s through all paylos s through each payl 5304, 107 Sat ar	ad sets load set in tun r1/537.35	n, so	Add § Clear § Auto § Refresh
	00	€ →	Search											0 matches	Clear
	2 payload	positions												Length: 653	

3. Configure Intruder Settings: In the Intruder tab, select the Cluster Bomb attack type. Set the payload position to the desired parameter and load the payload list. Customize any other attack options, such as payload processing rules or response handling, based on usersr requirements.

	truder Repea											
board Ta	arget Prov	/ Intruder	Repeater	Sequencer	Decoder	Comparer	Logger	Extender	Project options	User options	Learn	
2 ×	+											Q
ions Pavi	loads Res	surce Pool	Options									
	nodified baselie	e request										
I the dards	al-of-service m	via ino rasultsi										
Store full		All plotteration										
Store full	paycoads											
	te berne en bekenne	-		pecified expression	XNS.							
🗸 Flag resu	It items with re	sponses match	ing these expre		ons.							
		sponses match			XIIS.							
Paste	It items with re	sponses match			XNS.							
	error	sponses match			xns.							
Paste	error exception illegal invalid	sponses match			xns.							
Paste Load Remove	error exception illegal invalid fail	sponses match			xıs.							
Paste Load	error exception illegal invalid fail stack	sponses match			xns.							
Paste Load Remove	error exception illegal invalid fail stack access	sponses match			xns.							
Paste Load Remove	error exception illegal invalid fail stack	sponses match			ms.							
Paste Load Remove	error exception illegal invalid fail stack access				ms.							
Paste Load Remove Clear Add	error exception illegal imalid fail stack access directory Enter a new	lem			ns.							
Paste Load Remove Clear Add Match type:	error exception illegal invalid fail stack access directory Enter a new Simple striv	lem			ms.							
Paste Load Remove Clear Add Match type:	error exception illegal imalid fail stack access directory Enter a new	lem			ms.							
Paste Load Remove Clear Add Match type:	error exception lilegal invalid fail stack access directory Enter a new Simple strift Regex	lem			ms.							



4. Fine-tune Attack Options: Adjust the attack options as needed. Users can specify the number of threads or iterations to control the attack speed and intensity. It's important to strike a balance between thoroughness and performance to avoid overwhelming the target server.

Burp Suite Community Edition v2022.9.6 - Temp	rary Project 💿 🔘 🗧
Burp Project Intruder Repeater Window Help	
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Logger	Extender Project options User options Learn
1 × 2 × +	Q, 1
Positions Payloads Resource Pool Options	
Save Options [Proversion only] Find out more	Start attack
This setting allows you to save your attack to the current project file. The attack will then be available from the Dashb	ard whenever you open this project.
Save attack to project file	
Classe areas or bedress one	

5. Launch the Attack: Start the Cluster Bomb attack and monitor the progress in real-time. Burp Suite Intruder will iterate through each payload combination and send requests to the target, capturing the responses.

Results	Positions Payloads	Resource Pool Options							
Filter: Show	ing all items								(
Request \land	Payload 1	Payload 2	Status	Error	Timeout	Length	error	exception	illega
)			302			253			
	admin	admin	302			253			
2	user1	admin	302			253			
3	user2	admin	302			253			
4	aadmin	admin	302			253			
5	newuser	admin	302			253			
5	user3	admin	302			253			
7	test	admin	302			253			
3	admin	password	302			253			
9	user1	password	302			253			
0	user2	password	302			253			
1	aadmin	password	302			253			
12	newuser	password	302			253			
13		anonicard	202			252			

6. Analyze the Results: Once the attack is complete, review the responses to identify any successful hits or potential vulnerabilities. Pay attention to any unexpected behaviors or error messages that could indicate a successful exploitation.

	/e Columns								
Results	Positions Payloads	Resource Pool Options							
Filter: Showi	ing all items								(
Request \land	Payload 1	Payload 2	Status	Error	Timeout	Length	error	exception	illega
23	duiiiii	duilli 125	302	Ц		233			
30	user1	admin123	302			253			
31	user2	admin123	302			253			
32	aadmin	admin123	302			253			
33	newuser	admin123	302			253			
34	user3	admin123	302			253			
35	test	admin123	302			253			
36	admin	test	302			253			
37	user1	test	302			253			
38	user2	test	302	Ē	Ē	253			
39	aadmin	test	302	ī	ñ	253			
40	newuser	test	302	l ñ		253			
41	user3	test	302	ň		253			
42	test	test	200	i n		6191	1		

7. Refine and Repeat: Based on the results, refine users' payload list, attack options, or parameters to further optimize the Cluster Bomb attack. Iteratively repeating the process can help uncover additional weaknesses or strengthen the overall security of the web application.

4. CONCLUSION

In conclusion, this research paper has thoroughly examined the significance of Burp Suite tools, with a specific focus on Burp Repeater and Burp Intruder, in the realm of web application security testing. The study has elucidated the practical application of Burp Suite tools through controlled experiments, emphasizing their utility in uncovering vulnerabilities while adhering to OWASP secure coding practices. Burp Repeater emerges as a tool that enhances testing efficiency by providing essential



E-ISSN: 2582-2160 • Website: <u>www.ijfmr.com</u> • Email: editor@ijfmr.com

functions such as request saving, reloading, request and answer comparison, and bookmarking. The paper underscores the value of Burp Repeater in facilitating a streamlined and effective testing process.

Furthermore, the research delves into the capabilities of Burp Intruder, showcasing its role in automating and streamlining various attack types on web applications. This tool proves instrumental in identifying vulnerabilities and weaknesses, offering extensive options for payload customization, response handling, and attack configuration. The multiple attack types supported by Burp Intruder contribute to its versatility and effectiveness in assessing web application security.

The overarching objective of this study is to enhance the understanding and utilization of Burp Suite tools, aligning them with the principles of OWASP secure coding practices. By employing Burp Suite tools, organizations can not only discover vulnerabilities but also enhance their overall security posture and ensure the reliability of their web applications. This research serves as a valuable guide for security practitioners and organizations seeking to leverage Burp Suite tools effectively in the dynamic landscape of web application security testing.

5. **References**

- 1. Open Web Application Security Project (OWASP). "OWASP Secure Coding Practices Quick Reference Guide." [Online] Available: https://owasp.org/www-pdfarchive/OWASP_SCP_Quick_Reference_Guide_v2.pdf
- 2. Open Web Application Security Project (OWASP). "Vulnerability Scanning Tools." [Online] Available: https://owasp.org/www-community/Vulnerability_Scanning_Tools
- 3. Kali Linux. "Tool Documentation." [Online] Available: www.kali.org/tools/burpsuite/
- 4. PortSwigger. "Burp Suite documentation." [Online] Available: https://portswigger.net/burp/documentation
- 5. PortSwigger. "Burp Suite Intruder Attacks." [Online] Available: https://portswigger.net/burp/documentation/desktop/tools/intruder/configure-attack/attack-types
- O. B. Fredj, O. Cheikhrouhou, M. Krichen, H. Hamam, A. Derhab. "An OWASP Top Ten Driven Survey on Web Application Protection Methods." Institute of Electrical and Electronics Engineers (IEEE), 2020. [Online] Available: <u>https://www.researchgate.net/publication/346035890 An OWASP Top Ten Driven Survey on</u> Web_Application_Protection_Methods
- J. Warsinkse. "CISSP: Certified Information Systems Security Professional." Wiley, 2019. [Online] Available: https://onlinelibrary.wiley.com/doi/book/10.1002/9781119423300
- 8. I. Babincev, D. Vuletic. "Web application security analysis using the Kali Linux operating system." Vojnotehnicki glasnik, 2016. [Online] Available: https://www.redalyc.org/pdf/6617/661770082010.pdf
- 9. Secure Coding. [Online] Available: https://www.securecoding.org/ [10] "How to Use Burp Suite's Intruder Tool to Pentest Web Apps." MakeUseOf. [Online] Available: https://www.makeuseof.com/how-to-use-burp-intruder/
- 10. "Burp for Beginners: How to Use Repeater." YouTube. [Online] Available: https://www.youtube.com/watch?v=LfuU5u6C2Sk



E-ISSN: 2582-2160 • Website: www.ijfmr.com • Email: editor@ijfmr.com

- 11. D. Assaf. "An Approach Toward Implementing Continuous Security in Agile Environment." Instituto Politecnico do Porto (Portugal), 2022. [Online] Available: http://hdl.handle.net/10400.22/16233
- 12. G. Khawaja. "Web Penetration Testing and Secure Software Development Lifecycle." In Kali Linux Penetration Testing Bible, Wiley, 2021, pp.231-255. [Online] Available: https://ieeexplore.ieee.org/document/9936716/
- A. Al Anhar, Y. Suryanto. "Evaluation of Web Application Vulnerability Scanner for Modern Web Application." 2021 International Conference on Artificial Intelligence and Computer Science Technology (ICAICST), Yogyakarta, Indonesia, 2021. doi: 10.1109/ICAICST53116.2021.9497831. [Online] Available: https://ieeexplore.ieee.org/document/9497831/
- 14. A. Mohan, G. A. Swaminathan, N. J. Shafana. "Automated Tools and Techniques in Vulnerability Assessment." 2022 4th International Conference on Smart Systems and Inventive Technology (ICSSIT), Tirunelveli, India, 2022. doi: 10.1109/ICSSIT53264.2022.9716474. [Online] Available: https://ieeexplore.ieee.org/document/9716474/