



HOME

ABOUT

CONTENT

OTHERS

WEB APPLICATION SECURITY TESTING PROJECT

Manual Vulnerability Assessment
using Burp Suite





HOME

ABOUT

CONTENT

OTHERS

PROJECT OVERVIEW

PERFORMED
MANUAL WEB
APPLICATION
SECURITY
TESTING

IDENTIFIED
COMMON
VULNERABILITIE
S SUCH AS XSS
AND SQL
INJECTION

TESTING
CONDUCTED IN
A CONTROLLED
AND ETHICAL
ENVIRONMENT

TARGET APPLICATION:

testphp.vulnweb.com (intentionally vulnerable test site)





TOOLS & METHODOLOGY

The screenshot shows the Burp Suite interface. At the top is a table of network traffic with columns: #, Host, Method, URL, Params, Edited, Status code, Length, MIME type, Extension, Title, Notes, TLS, and IP. Below this is a list of 13 requests, all GETs to 'http://testphp.vulnweb.com/search.php' with status 200, length 4954, and type HTML. The 'Request' panel shows a Pretty-printed version of a GET /search.php HTTP/1.1 request with headers Host, Accept-Language, Upgrade-Insecure-Requests, and User-Agent. The 'Response' panel shows a Pretty-printed version of an HTTP/1.1 200 OK response with headers Server, Date, Content-Type, and Connection.

#	Host	Method	URL	Params	Edited	Status code	Length	MIME type	Extension	Title	Notes	TLS	IP
1	http://testphp.vulnweb.com	GET	/			200	5180	HTML		Home of Acunetix Art		44.22	
2	http://testphp.vulnweb.com	GET	/search.php			200	4954	HTML	php	search		44.22	
3	http://testphp.vulnweb.com	GET	/search.php			200	4954	HTML	php	search		44.22	
4	http://testphp.vulnweb.com	GET	/search.php			200	4954	HTML	php	search		44.22	
5	http://testphp.vulnweb.com	GET	/search.php			200	4954	HTML	php	search		44.22	
6	http://testphp.vulnweb.com	GET	/search.php			200	4954	HTML	php	search		44.22	
7	http://testphp.vulnweb.com	GET	/search.php			200	4954	HTML	php	search		44.22	
8	http://testphp.vulnweb.com	GET	/search.php			200	4954	HTML	php	search		44.22	
9	http://testphp.vulnweb.com	GET	/search.php			200	4954	HTML	php	search		44.22	
10	http://testphp.vulnweb.com	GET	/search.php			200	4954	HTML	php	search		44.22	
11	http://testphp.vulnweb.com	GET	/search.php			200	4954	HTML	php	search		44.22	
12	http://testphp.vulnweb.com	GET	/search.php			200	4954	HTML	php	search		44.22	
13	http://testphp.vulnweb.com	GET	/search.php			200	4954	HTML	php	search		44.22	

TOOLS

- Burp Suite (Intercept, Repeater)

METHODOLOGY

- Intercept HTTP requests
- Analyze parameters and responses
- Inject payloads manually
- Validate vulnerabilities via server response



[HOME](#)[ABOUT](#)[CONTENT](#)[OTHERS](#)

IDENTIFIED VULNERABILITY: REFLECTED XSS

VULNERABILITY TYPE:
REFLECTED CROSS-SITE
SCRIPTING (XSS)

ENDPOINT: SEARCH.PHP

SEVERITY: MEDIUM

```
Pretty Raw Hex
1 POST /search.php?test=query HTTP/1.1
2 Host: testphp.vulnweb.com
3 Content-Type: application/x-www-form-urlencoded
4 Content-Length: 39
5 Connection: keep-alive
6
7 searchFor=<script>alert(1)</script>
8
9
```





XSS PROOF OF CONCEPT

USER INPUT IS
REFLECTED DIRECTLY
INTO THE HTML
RESPONSE

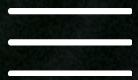
NO INPUT SANITIZATION
OR OUTPUT ENCODING
OBSERVED

INJECTED JAVASCRIPT
PAYLOAD IS RENDERED
IN THE BROWSER
CONTEXT

Response

	Pretty	Raw	Hex	Render
53				<!-- begin content -->
54				<!-- InstanceBeginEditable name="content_rgn" -->
55				<div id="content">
56				<h2 id='pageName'>
57				searched for: <script>
				alert(1)
				</script>
58				
59				</h2>
60				</div>
61				<!-- InstanceEndEditable -->
				<!--end content -->



[HOME](#)[ABOUT](#)[CONTENT](#)[OTHERS](#)

SQL INJECTION EVIDENCE

GET /listproducts.php?cat=1' HTTP/1.1

Error: You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '' at line 1

SINGLE QUOTE (')
CAUSED DATABASE
ERROR

INDICATES UNSANITIZED
INPUT PASSED TO SQL
QUERY

DATABASE ERROR
MESSAGES EXPOSED TO
USER



[HOME](#)[ABOUT](#)[CONTENT](#)[OTHERS](#)

MITIGATION & RECOMMENDATIONS

- Validate and sanitize all user input on the server side
- Use parameterized queries to prevent SQL Injection
- Encode user input before rendering it in HTML to mitigate XSS
- Conduct regular security testing during development



[HOME](#)[ABOUT](#)[CONTENT](#)[OTHERS](#)

SECURITY IMPACT

The identified vulnerabilities allow attackers to inject malicious scripts and manipulate backend database queries.

If exploited in a real-world environment, these issues could lead to data leakage, session hijacking, and unauthorized system access.



[HOME](#)[ABOUT](#)[CONTENT](#)[OTHERS](#)

CONCLUSION

This project demonstrates hands-on experience in manual web application security testing.

Through HTTP request analysis and controlled payload injection, multiple security vulnerabilities were identified in a deliberately vulnerable environment.

The testing process emphasizes ethical testing practices and responsible security research.



[HOME](#)[ABOUT](#)[CONTENT](#)[OTHERS](#)

KEY TAKEAWAYS

- Gained hands-on experience in manual web security testing
- Learned to analyze HTTP requests and responses using Burp Suite
- Understood how improper input handling leads to XSS vulnerabilities
- Applied ethical and responsible testing practices





HOME

ABOUT

CONTENT

OTHERS

THANK YOU

THANK YOU FOR REVIEWING THIS PROJECT

THIS PROJECT REFLECTS MY INTEREST IN WEB APPLICATION SECURITY AND CONTINUOUS LEARNING IN CYBERSECURITY