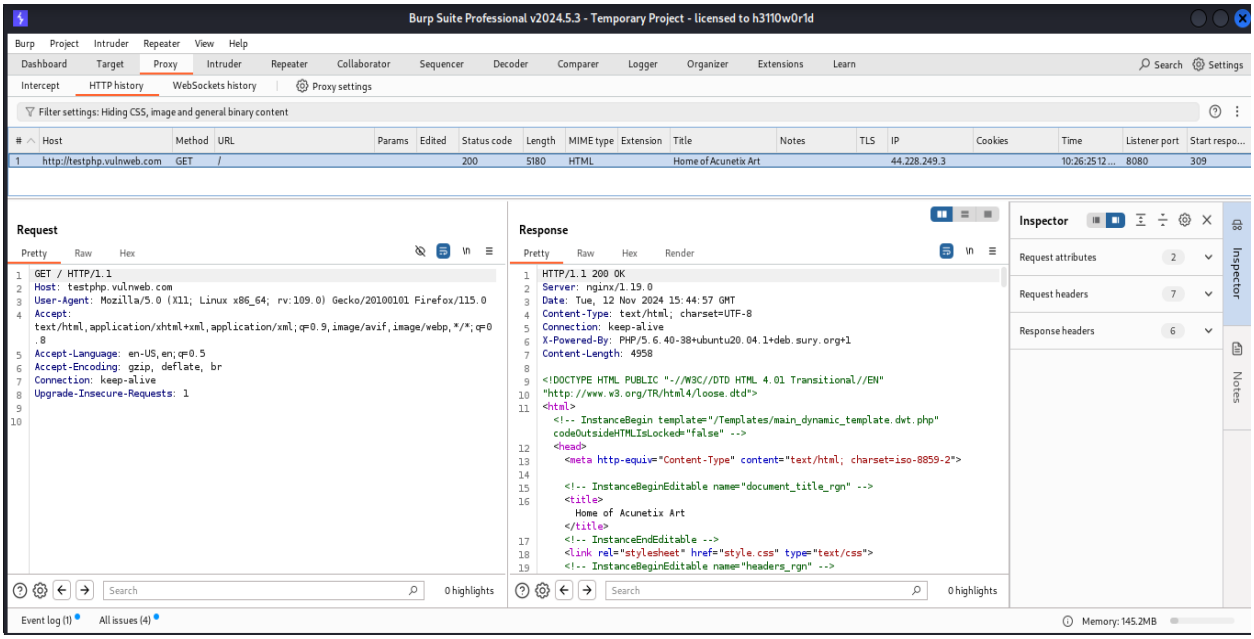
Lab 8: Web Application Security Testing with Burp Suite

Exercise 1:

• Document the HTTP request and response headers for the home page of the target application. What information do you find in these headers?

Seven 7 Request headers and Six 6 Response header were observed, the Request contain GET / HTTP/1.1, Host: testphp.vulnweb.com, User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0, Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8, Accept-Language: en-US,en;q=0.5, Accept-Encoding: gzip, deflate, br, Connection: keep-alive, Upgrade-Insecure-Requests: 1.

The Response to the contained HTTP/1.1 200 OK, Server: nginx/1.19.0, Date: Tue, 12 Nov 2024 15:44:57 GMT, Content-Type: text/html; charset=UTF-8, Connection: keep-alive, X-Powered-By: PHP/5.6.40-38+ubuntu20.04.1+deb.sury.org+1, Content-Length: 4958



Exercise 2:

• List the URLs discovered during the spidering process. Did you find any hidden or interesting pages?

Below are the discovered URL

http://testphp.vulnweb.com/AJAX/index.php

http://www.acunetix.com/

http://www.acunetix.com/

http://www.eclectasy.com/

http://www.eclectasy.com/Fractal-Explorer

http://www.eclectasy.com/Fractal-Explorer/index.html

http://download.macromedia.com/

http://download.macromedia.com/pub

http://download.macromedia.com/pub/shockwave

http://download.macromedia.com/pub/shockwave/cabs

http://download.macromedia.com/pub/shockwave/cabs/flash

http://download.macromedia.com/pub/shockwave/cabs/flash/swflash.cab

http://www.macromedia.com/

http://www.macromedia.com/shockwave

http://www.macromedia.com/shockwave/download

http://www.macromedia.com/shockwave/download/index.cgi

http://www.macromedia.com/shockwave/download/index.cgi?P1_Prod_Version=ShockwaveFlash

https://testphp.vulnweb.com/

https://testphp.vulnweb.com/

https://testphp.vulnweb.com/robots.txt

http://www.w3.org/

http://www.w3.org/1999

http://www.w3.org/1999/xhtml

http://www.w3.org/TR

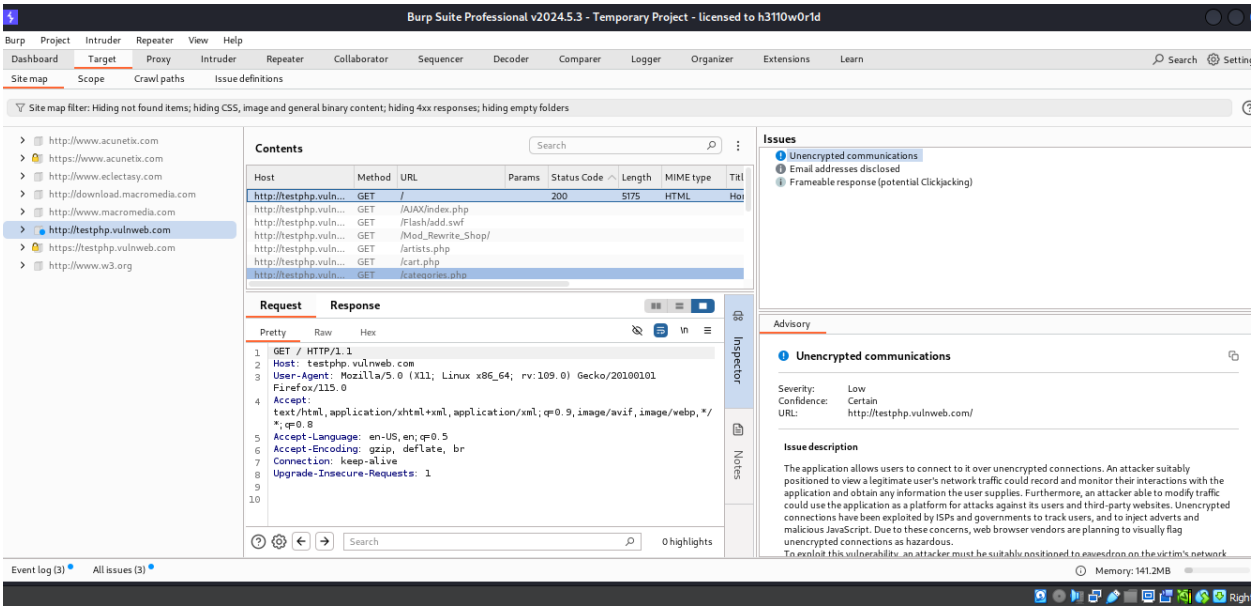
http://www.w3.org/TR/html4

http://www.w3.org/TR/html4/loose.dtd

http://www.w3.org/TR/xhtml1

http://www.w3.org/TR/xhtml1/DTD

http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd



Exercise 3:

• What vulnerabilities were detected by Burp Suite? Choose one vulnerability and explain how it could be exploited

These are the vulnerabilities found

1. Email addresses disclosed: Issue detail

The following email address was disclosed in the response: wvs@acunetix.com

2. Unencrypted communication
3. Frameable response (potential Clickjacking)

Email addresses disclosed

The presence of email addresses within application responses does not necessarily constitute a security vulnerability. Email addresses may appear intentionally within contact information, and many applications (such as web mail) include arbitrary third-party email addresses within their core content.

However, email addresses of developers and other individuals (whether appearing on-screen or hidden within page source) may disclose information that is useful to an attacker; for example, they may represent usernames that can be used at the application's login, and they may be used in social engineering attacks against the organization's personnel. Unnecessary or excessive disclosure of email addresses may also lead to an increase in the volume of spam email received.

Exercise 4:

• Capture and analyze the traffic with OWASP ZAP. What differences do you notice compared to Burp Suite?

OWASP ZAP automatically capture all the URL on the website visited without intercept the network and recorded them, this also includes Alert while Burp suite need to be intercept before the capture the request and response of the network, Moreso, spider scanning will be perform on Burp before one can see all the URL on the site and without active scanning vulnerabilities can not be found in Burp suite.

Exercise 5:

• Review the vulnerabilities identified by OWASP ZAP. Which tools detected the same vulnerabilities? What are the potential impacts of these vulnerabilities?

Eleven (11) Vulnerabilities were found, one of them is Absence of Anti-CSRF Tokens:

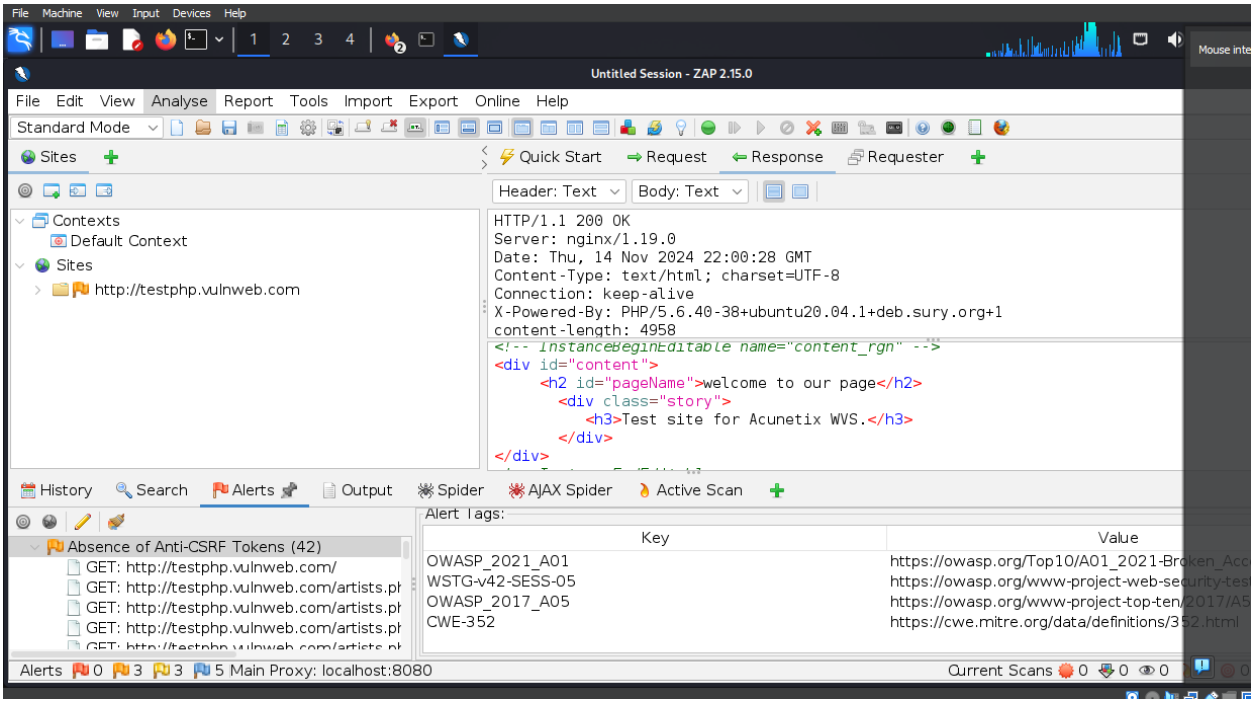
No Anti-CSRF tokens were found in a HTML submission form.

CSRF attacks are effective in several situations, including:

* The victim has an active session on the target site.

* The victim is authenticated via HTTP auth on the target site.

* The victim is on the same local network as the target site.



Mitigations:

1. Architecture and Design

Use a vetted library or framework that does not allow this weakness to occur or provides constructions that make this weakness easier to avoid.

For example, use anti-CSRF packages such as the OWASP CSRFGuard.

2. Implementation

Ensure that your application is free of cross-site scripting issues, because most CSRF defenses can be bypassed using attacker-controlled script.

Exercise 6:

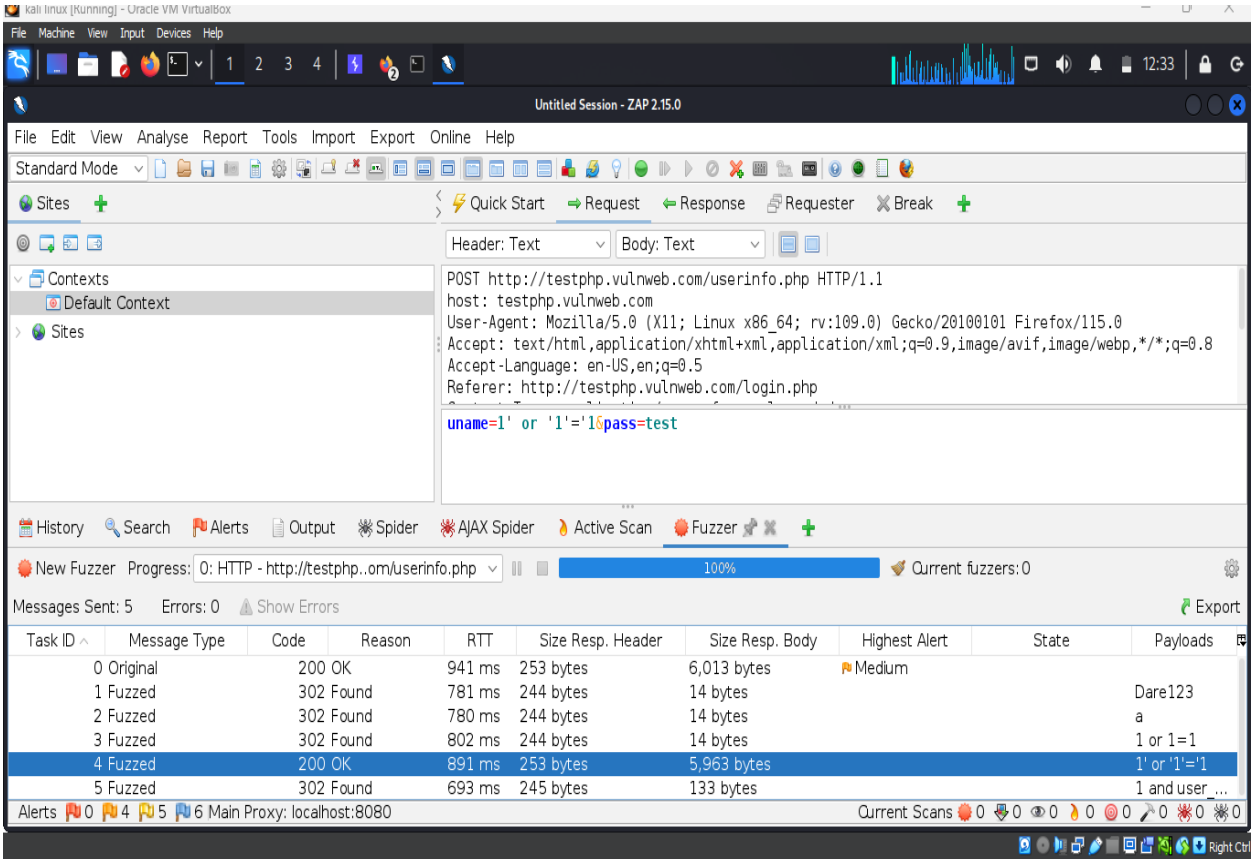
• Compare the findings of OWASP ZAP with Burp Suite. Which tool provided more detailed information? Which tool do you prefer for vulnerability scanning? Why?

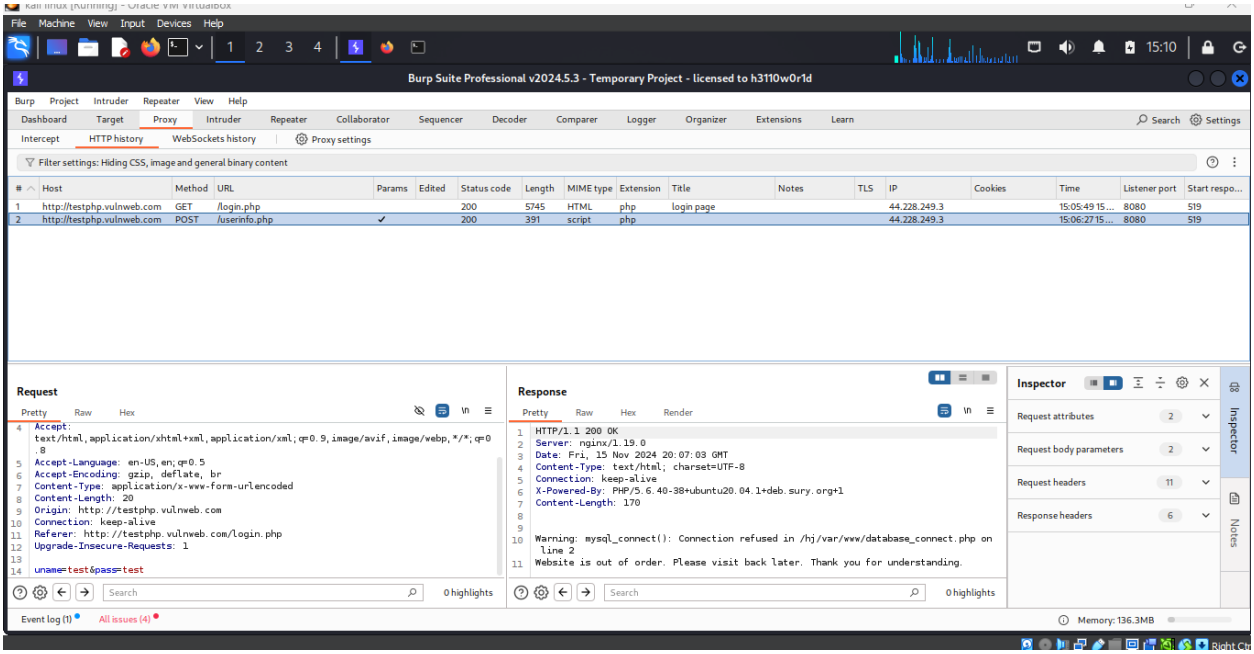
OWASP ZAP provides more detailed information about the targets sit than Burp Suite, And I prefer OWASP ZAP for vulnerability scanning than Burp suite because it details is findings, and you don't have to perform vulnerability scan separately unlike Burp suite a vulnerability scan is done separately. Also, OWASP ZAP is easy to navigate.

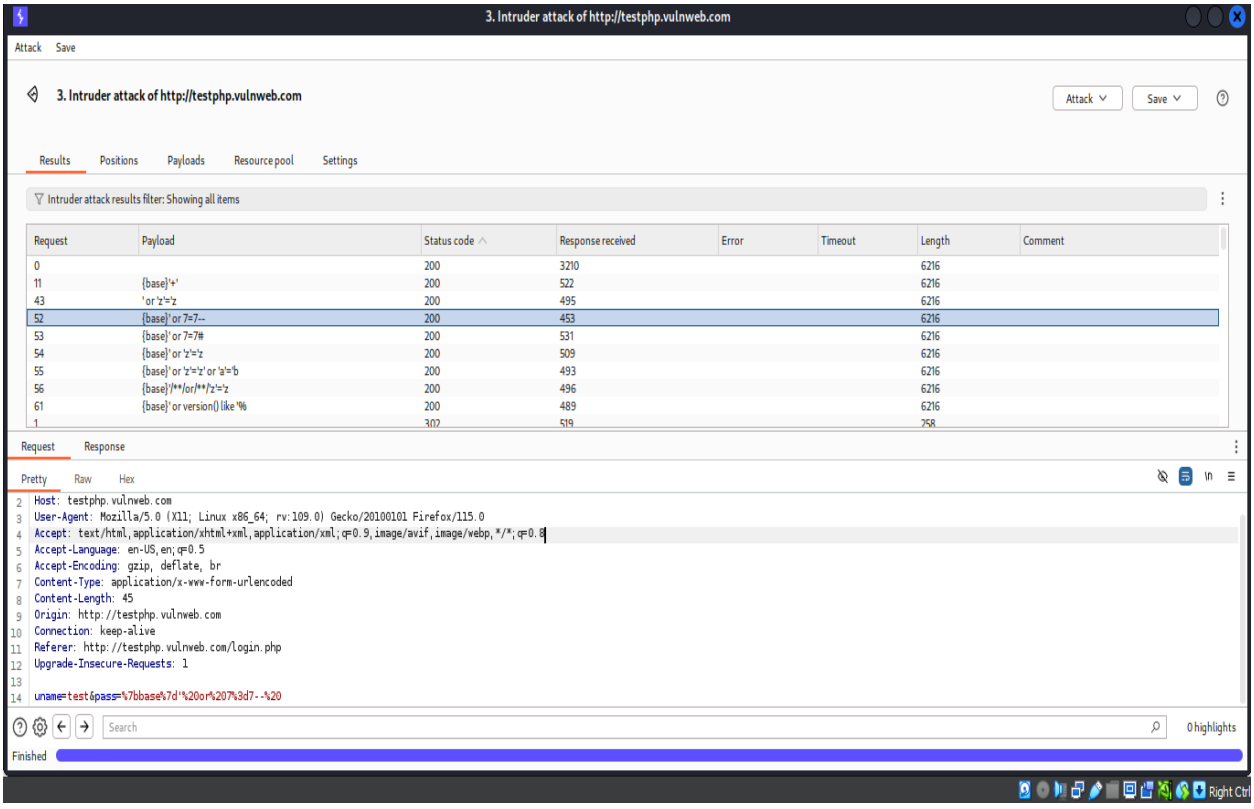
Exercise 7:

• Document any successful injections or errors encountered during fuzzing. What techniques were effective?

Both techniques were effective. The SQL injection is successfully injected.




Burp Suite Professional v2024.5.3 - Temporary Project - licensed to h3110w0r1d
Request
Response
Inspector
Request attributes
Request body parameters
Request headers
Response headers
Warning: mysql_connect(): Connection refused in /hj/var/www/database_connect.php on line 2
Website is out of order. Please visit back later. Thank you for understanding.


3. Intruder attack of http://testphp.vulnweb.com
Results
Positions
Payloads
Resource pool
Settings
Request
Response
Finished