

OWASP ZAP Project



OWASP
Zed Attack Proxy

TASK 1 Introduction and overview of the Tool

The tool was designed by the Open Web Application Security Project and called has OWASP ZAP [OWASP Foundation, the Open Source Foundation for Application Security | OWASP Foundation](#)

One of the best practices for protecting your web server and the clients accessing your websites is to find the vulnerabilities in your website

This task which can be automated with this tool "OWASP ZAP" from cross site scripting to sql injections covering all the top 10 vulnerabilities.

OWASP is a non-profit organization and online community that provides free content in the field of web security

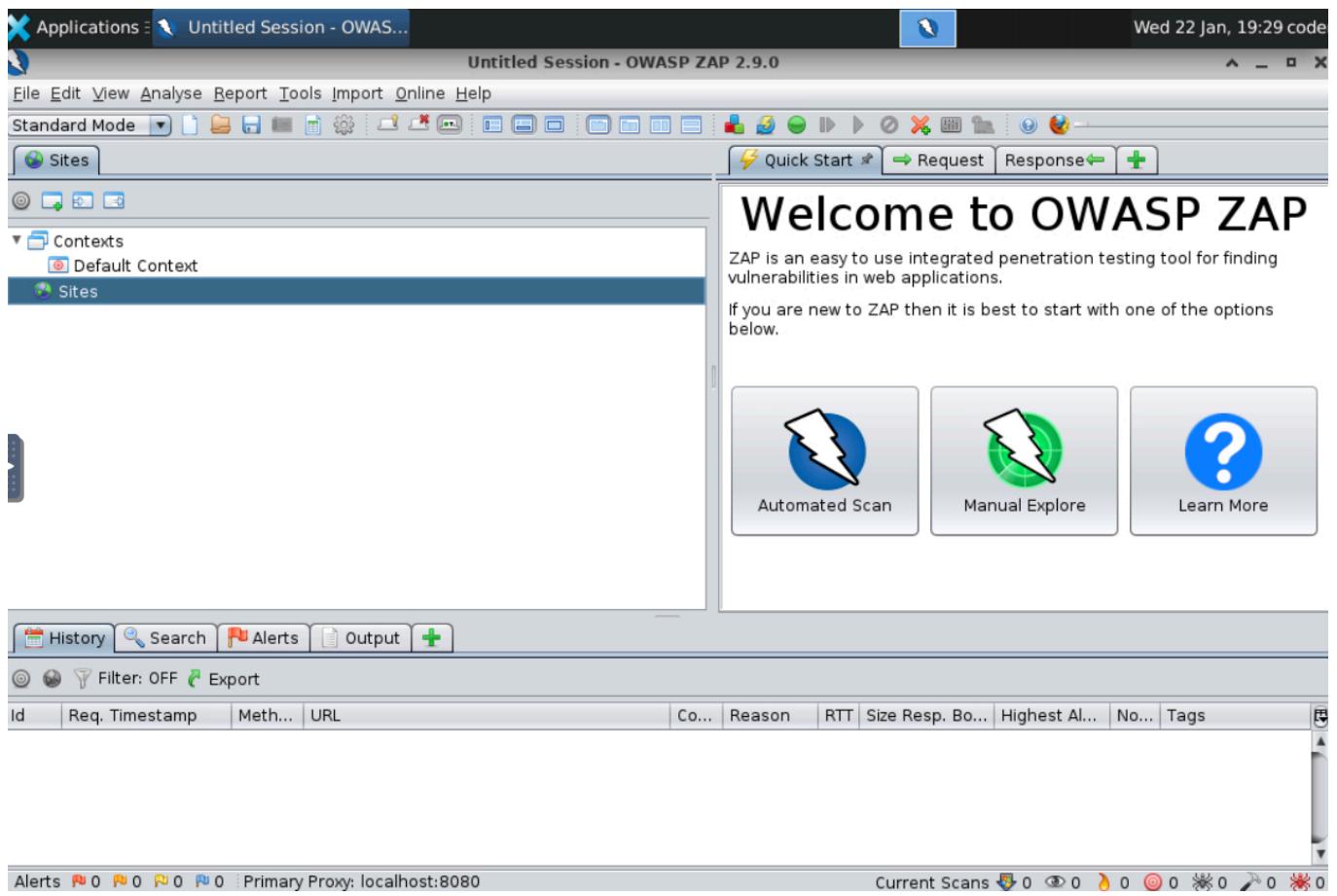
It is good for web security testing and which includes the intercepting proxy, active and passive vulnerability scanners a spider web crawler a buzzer and an http request sender and brute forcer and much more

This also has a scripting engine that can be used to automate activities or to create new functionality

But everything has a flaw a until it is found so here it can also generate false positives or miss critical vulnerabilities in some cases

Remember Automated scanning tools should not be used as a replacement for manual vulnerability exploitation but rather be used in conjunction alongside manual vulnerability exploitation and backup to more manual methods.

TASK 2 OWASP ZAP Layout and First Scan



This is the actual layout of the OWASP ZAP 2.9.0 version

we can see the sites tab which contains two: 1.Contexts which is a way of setting the url together and other 2: Sites which will be used for the websites that has been accessed and targeted by the ZAP

Quick start tab :

This is the easiest way to start using ZAP Which is for automated scanned tab when clicked we can see the url section which we need to provide for the tool or to the automated scan

The screenshot shows the OWASP ZAP interface with the 'Automated Scan' tab selected. At the top, there are tabs for 'Quick Start', 'Request', 'Response', and a green plus sign button. Below the tabs, the title 'Automated Scan' is displayed next to a blue lightning bolt icon. A descriptive text block explains how to launch an automated scan by entering a URL and pressing 'Attack'. It also includes a warning about attacking applications without permission.

This screen allows you to launch an automated scan against an application - just enter its URL below and press 'Attack'.

Please be aware that you should only attack applications that you have been specifically been given permission to test.

URL to attack:

Use traditional spider:

Use ajax spider: with

Progress: Not started

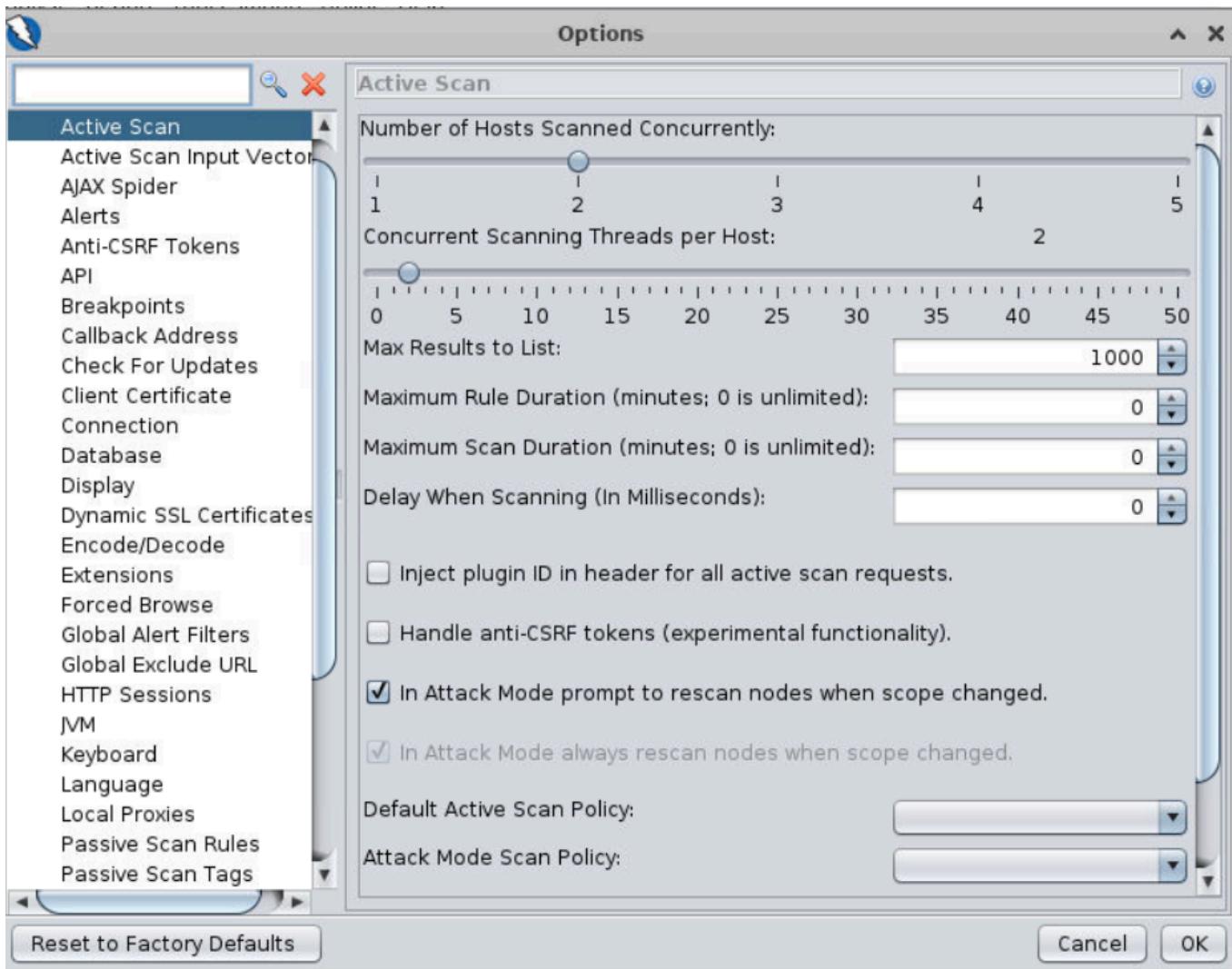
Along with this we can see other tabs which is History, Search, Alerts, Output

The screenshot shows the 'History' tab selected in the bottom navigation bar. The main area displays a table of network requests with columns for Id, Req. Timestamp, Meth..., URL, Co..., Reason, RTT, Size Resp. Body, Highest Al..., No..., and Tags. At the bottom, there are summary counts for Alerts, Current Scans, and various exploit types.

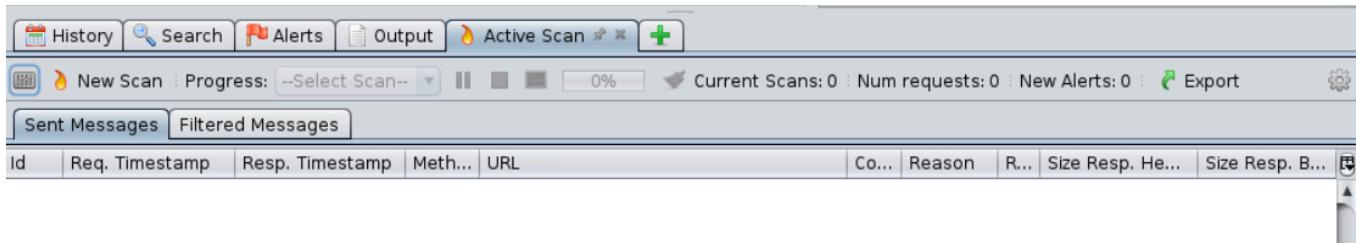
History where we can see the results of the scan in live like what it is scanning the id, Req. Timestamp Method and URI, Code, Method, Reason, RTT , Size Resp.Body, Highest Alert, Note, Tags

The Options button allows us to change the settings of the program

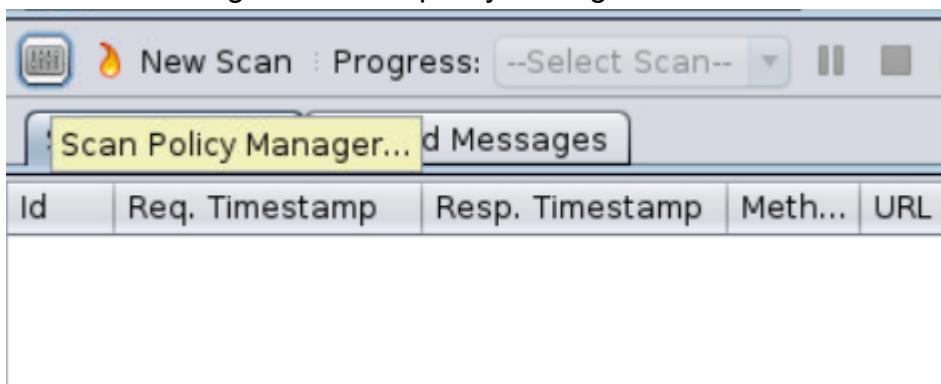




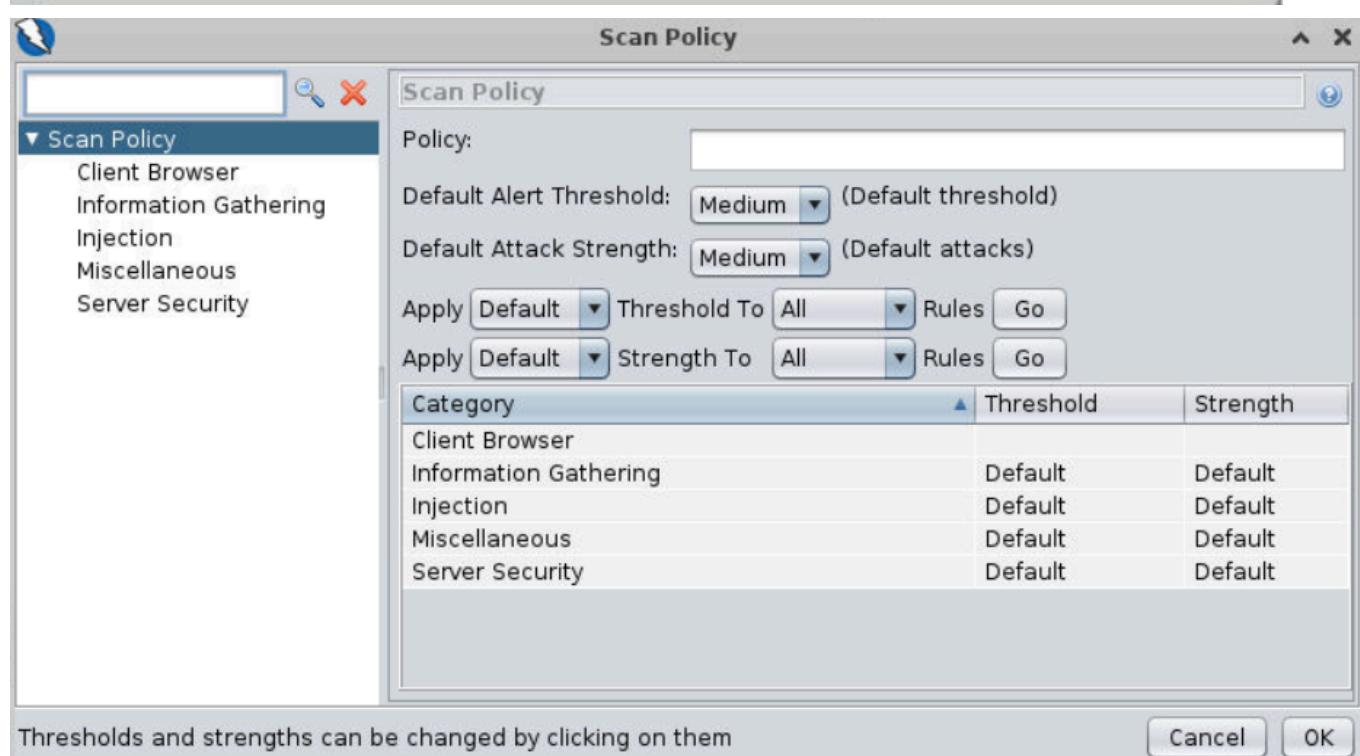
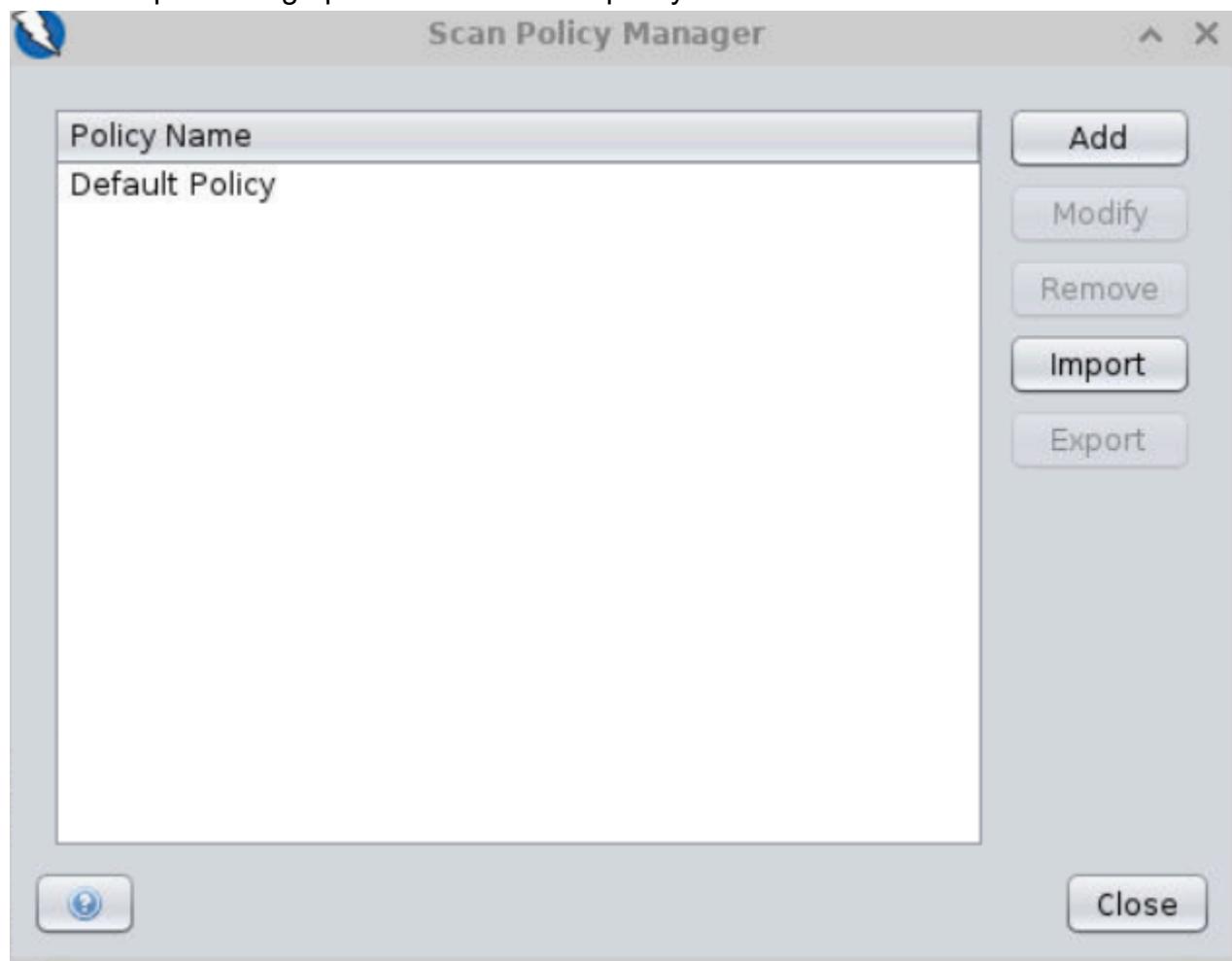
We can also modify the scan policy by adding the active scan tab by clicking the "+"



and then clicking to the scan policy manager

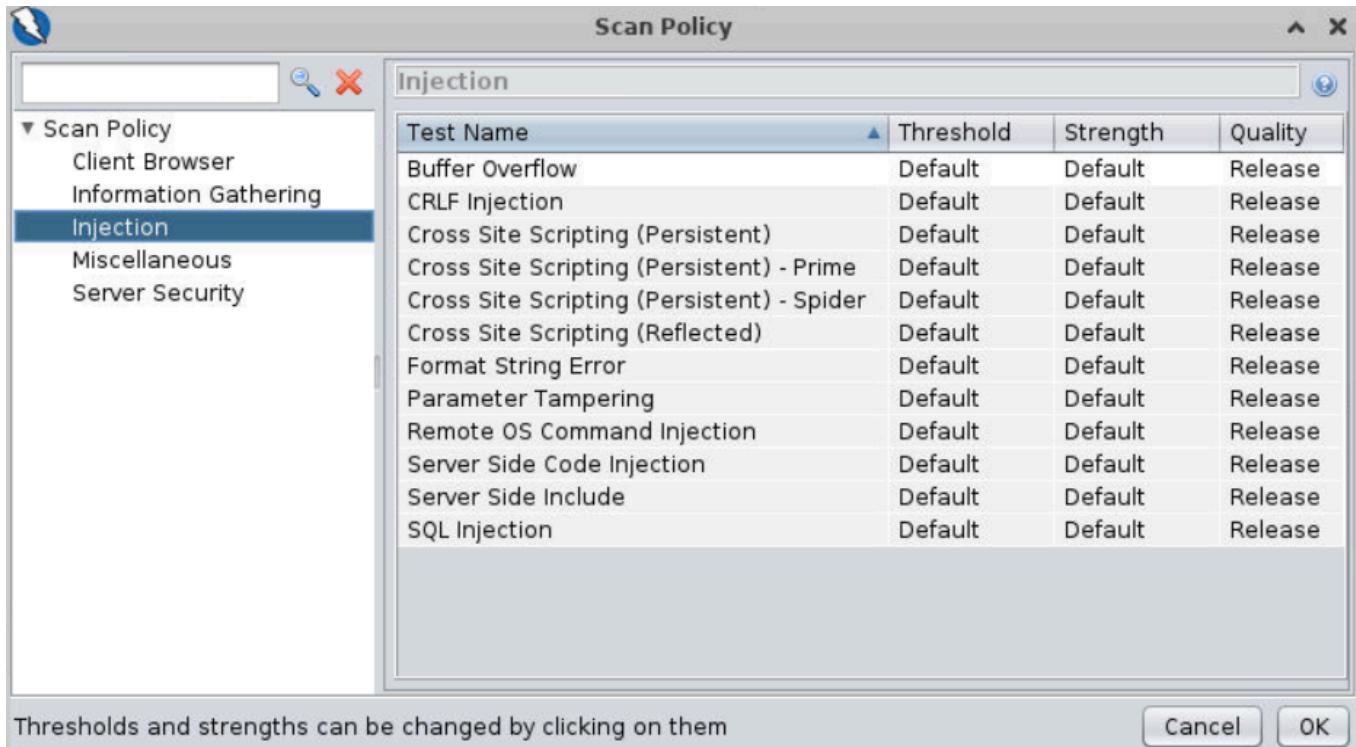


we will be presenting options with the scan policy button



Adding a scan policy looks like this where we can customize the policies which we want to go

for and setting out the threshold and strength



TASK 3 Analyzing the OWASP ZAP Scan Results and Generating Report

After the scan is done we can Get the report actually "cool right " so there is no need of doing a separate report and we can analyze it
we can see this in the sites tab

The screenshot shows the OWASP ZAP 2.9.0 interface. In the top left, there's a sidebar with 'Contexts' (Default Context) and 'Sites' (http://127.0.0.1). The main area is titled 'Automated Scan' with a sub-instruction: 'This screen allows you to launch an automated scan against an application - just enter its URL below and press 'Attack''. It also includes a note: 'Please be aware that you should only attack applications that you have been specifically been given permission to test.' Below these are fields for 'URL to attack' (http://127.0.0.1/mutillidae/) and 'Use traditional spider' (checked). At the bottom, there's a table of 'Sent Messages' with columns: Id, Req. Timestamp, Resp. Timestamp, Method, URL, Code, Reason, RTT, Size, Resp. Head., and Size, Resp. Body. The table lists 14 rows of network traffic. The bottom navigation bar includes tabs for History, Search, Alerts, Output, Active Scan, Spider, and a plus sign icon.

And in the below is the alerts tab where all the vulnerabilities are listed

The screenshot shows the 'Alerts' tab in OWASP ZAP. On the left, a tree view shows 'Alerts (14)' expanded, with categories like 'Cross Site Scripting (Reflected)' (2), 'SQL Injection' (2), 'X-Frame-Options Header Not Set' (32), etc. The right side contains descriptive text: 'Full details of any selected alert will be displayed here.', 'You can manually add alerts by right clicking on the relevant line in the history and selecting 'Add alert'.', and 'You can also edit existing alerts by double clicking on them.' Below this text, there's a list of alerts with colored flags: 2 red, 2 orange, and 10 yellow.

The alerts are listed has three different categories marking with different colored flags

- 1.High - Red flag
- 2.Medium - Orange flag
- 3.Low - Yellow flag

The screenshot shows the OWASP ZAP interface. The top navigation bar includes 'File', 'Edit', 'View', 'Analyse', 'Report', 'Tools', 'Import', 'Online', and 'Help'. Below the menu is a toolbar with icons for standard mode, sites, history, search, alerts, output, active scan, spider, and a plus sign. The main window has tabs for 'Sites' (selected), 'Quick Start', 'Request', 'Response', and 'Header: Text' and 'Body: Text' dropdowns.

The left sidebar displays a tree view of the application structure under 'http://127.0.0.1':

- images
 - GET:index.php
 - GET:mutillidae
- mutillidae
 - GET:mutillidae(page)
 - GET:robots.txt
 - GET:sitemap.xml
- webservices

The 'Alerts' tab is selected, showing 14 alerts:

- Cross Site Scripting (Reflected) (2)
- SQL Injection (2)
- X-Frame-Options Header Not Set (32)
- Absence of Anti-CSRF Tokens (5)
- Application Error Disclosure
- Cookie No HttpOnly Flag (3)

The 'Cross Site Scripting (Reflected)' alert details are shown in the right panel:

Cross Site Scripting (Reflected)

URL: http://127.0.0.1/mutillidae/hints-page-wrapper.php?levelHintIncludeFile=%3C%2Ftd%3E%3Cscript%3Ealert%281%29%3B%3C%2Fscript%3E%3Ctd%3E

Risk: High

Confidence: Medium

Parameter: levelHintIncludeFile

Attack: </td><script>alert(1);</script><td>

Evidence: </td><script>alert(1);</script><td>

CWE ID: 79

WASC ID: 8

Source: Active (40012 - Cross Site Scripting (Reflected))

Description: Cross-site Scripting (XSS) is an attack technique that involves echoing attacker supplied code into a user's browser instance. A browser instance

we can analyze the Vulnerabilities detailed

The Important thing which is the Reporting is also done by this tool has mentioned earlier

The screenshot shows the OWASP ZAP 2.9.0 interface. The top menu bar includes File, Edit, View, Analyse, Report, Tools, Import, Online, and Help. The Report tab is currently selected. The left sidebar shows a tree view of URLs and files, including 'Sites' and various 'GET' requests for 'http://127.0.0.1/mutillidae/'. The main pane displays an HTTP response header and body. The header shows 'HTTP/1.1 200 OK' and the body contains an error message from MySQL. The bottom pane shows a list of alerts, with 'Cross Site Scripting (Reflected)' selected, providing detailed information about the attack, including URL, risk level, and attack payload.

Saving has a Html file and then converting into to pdf and submitting to the application owners is a much important task in VAPT(Vulnerability Assessment and Penetration testing)

TASK 4 Setting Up Foxy Proxy in Firefox to use OWASP ZAP as a Proxy

This Feature is very Important for a Web Security Testing Tool

Foxy Proxy which enables us to intercept the traffic if i would say the request we wanted to send to the server will be not sent directly it is ,intercepted at us the address where the zap is listening "in this tool"

In order to

Owasp zap is considered an integrated non-transparent proxy and when using the proxy we need to configure the browser for the request to the ZAP tool

Two ways to configure the firefox browser manually or we can install a proxy Firefox add-on called FoxyProxy which requires an initial configuration

Applications Mozilla Firefox Untitled Session - OWAS... Thu 23 Jan, 14:37 coder

FoxyProxy Options 127.0.0.1/mutillidae/ +

127.0.0.1/mutillidae/ ⌂ 127

Some of Firefox's security features may offer less protection on your current operating system. Don't show again

It looks like you haven't started Firefox in a while. Do you want to clean it up for a fresh welcome back! Refresh Firefox...

OWASP Mutillidae II: Keep Coding!

Version: 2.11.15 Security Level: 0 (Hosed) Hints

Home | Login/Register | Toggle Hints | Toggle Security | Enforce TLS | Resources

OWASP 2017 OWASP 2013 OWASP 2010 OWASP 2007 Web Services Others Labs Documentation Resources

Hints and Videos

What Should I Do? Help Me!

Listing of vulnerabilities Video Tutorials

FoxyProxy

Disable 127.0.0.1:8081 8081

More Quick Add Exclude Host Set Tab Proxy Unset Tab Proxy Options Location IP Log

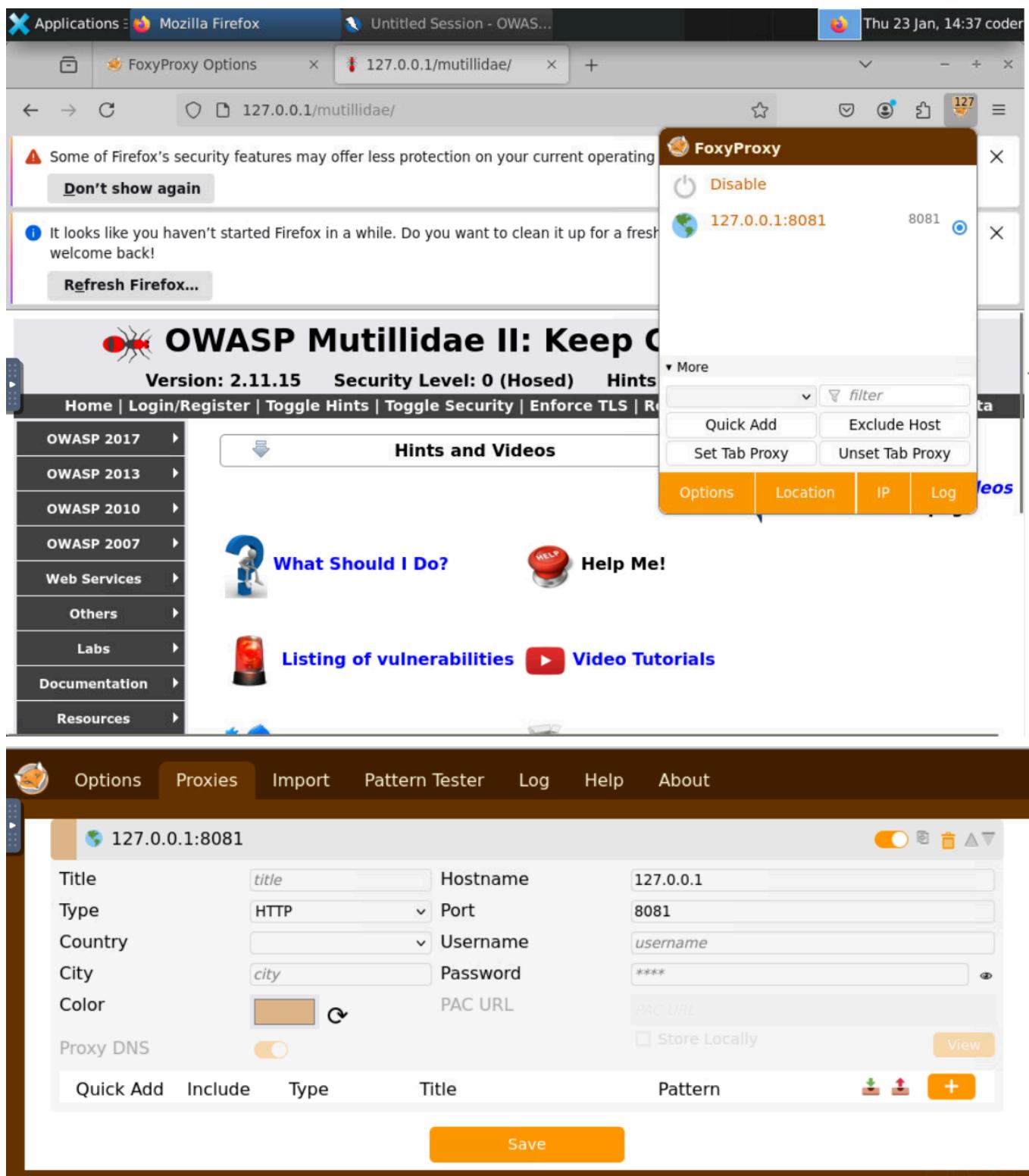
Options Proxies Import Pattern Tester Log Help About

127.0.0.1:8081

Title: title Hostname: 127.0.0.1
Type: HTTP Port: 8081
Country: Username: username
City: Password: ****
Color: PAC URL:
Proxy DNS: Store Locally View

Quick Add Include Type Title Pattern

Save



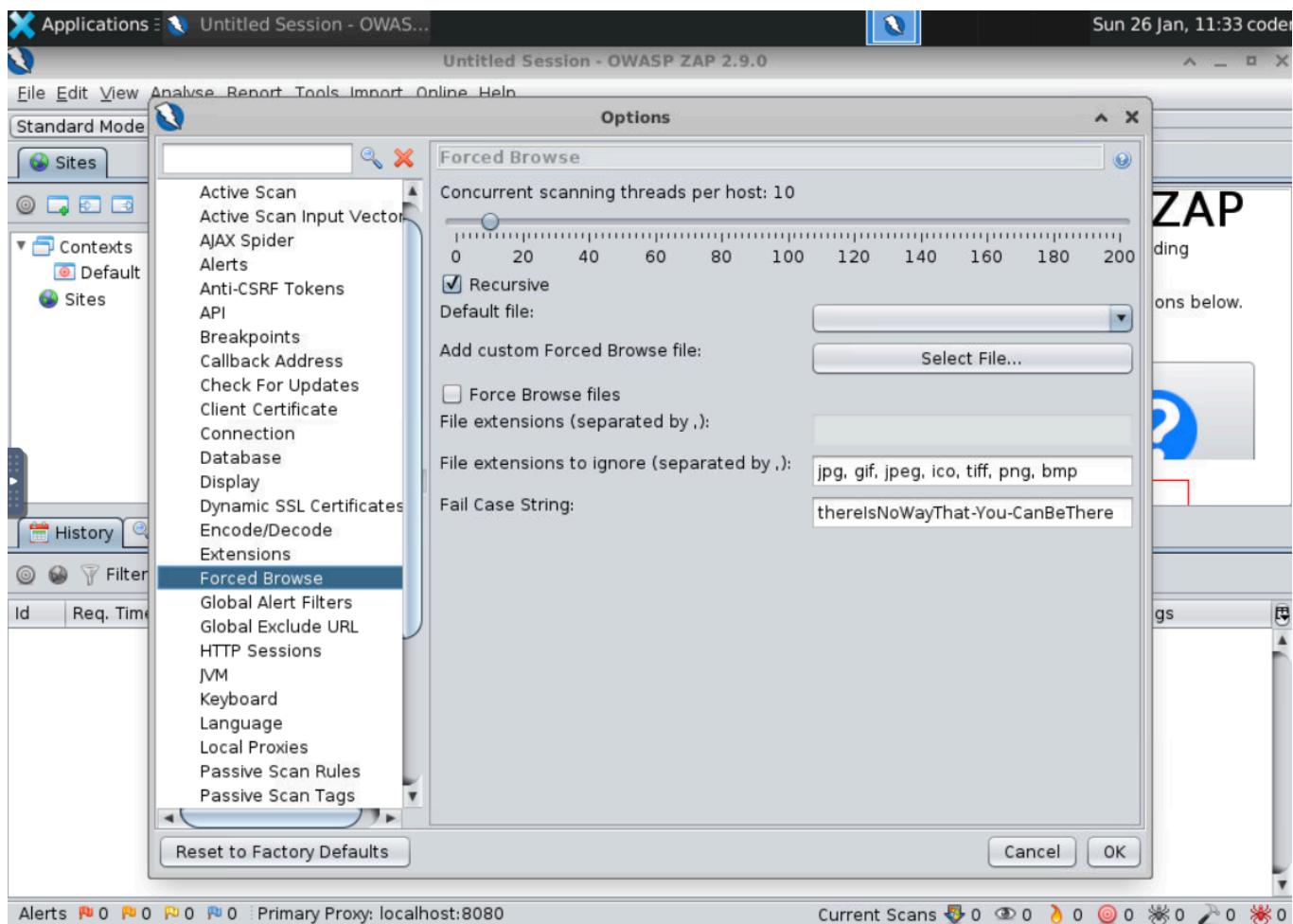
Id	Req. Timestamp	Meth...	URL	Co...	Reason	R...	Size	Resp. B...	Highest A...	N...	Tags
23	1/23/25, 2:36:2...	GET	http://127.0.0.1/mutillidae/javascript/i...	200	OK	1...	117 bytes		Low		
24	1/23/25, 2:36:2...	GET	http://127.0.0.1/mutillidae/javascript/i...	200	OK	1...	339 bytes		Low		
26	1/23/25, 2:36:2...	GET	http://127.0.0.1/mutillidae/javascript/i...	200	OK	2...	1,332 bytes		Low		Comment
27	1/23/25, 2:36:2...	GET	http://127.0.0.1/mutillidae/javascript/i...	200	OK	2...	1,488 bytes		Low		
28	1/23/25, 2:36:2...	GET	http://127.0.0.1/mutillidae/javascript/i...	200	OK	2...	535 bytes		Low		
38	1/23/25, 2:36:3...	GET	http://detectportal.firefox.com/canoni...	200	OK	1...	90 bytes		Medium		
40	1/23/25, 2:36:3...	GET	http://detectportal.firefox.com/succes...	200	OK	3...	8 bytes		Low		
41	1/23/25, 2:36:3...	GET	http://detectportal.firefox.com/succes...	200	OK	3...	8 bytes		Low		
49	1/23/25, 2:37:3...	GET	http://detectportal.firefox.com/canoni...	200	OK	1...	90 bytes		Medium		
50	1/23/25, 2:37:3...	GET	http://detectportal.firefox.com/succes...	200	OK	2...	8 bytes		Low		
51	1/23/25, 2:37:3...	GET	http://detectportal.firefox.com/succes...	200	OK	2...	8 bytes		Low		

Alerts 0 1 4 2 Primary Proxy: localhost:8081 Current Scans 0 0 0 0 0 0 0 0 0 0 0 0

TASK 5 Finding Files and Folders Using a Dictionary List Within OWASP ZAP

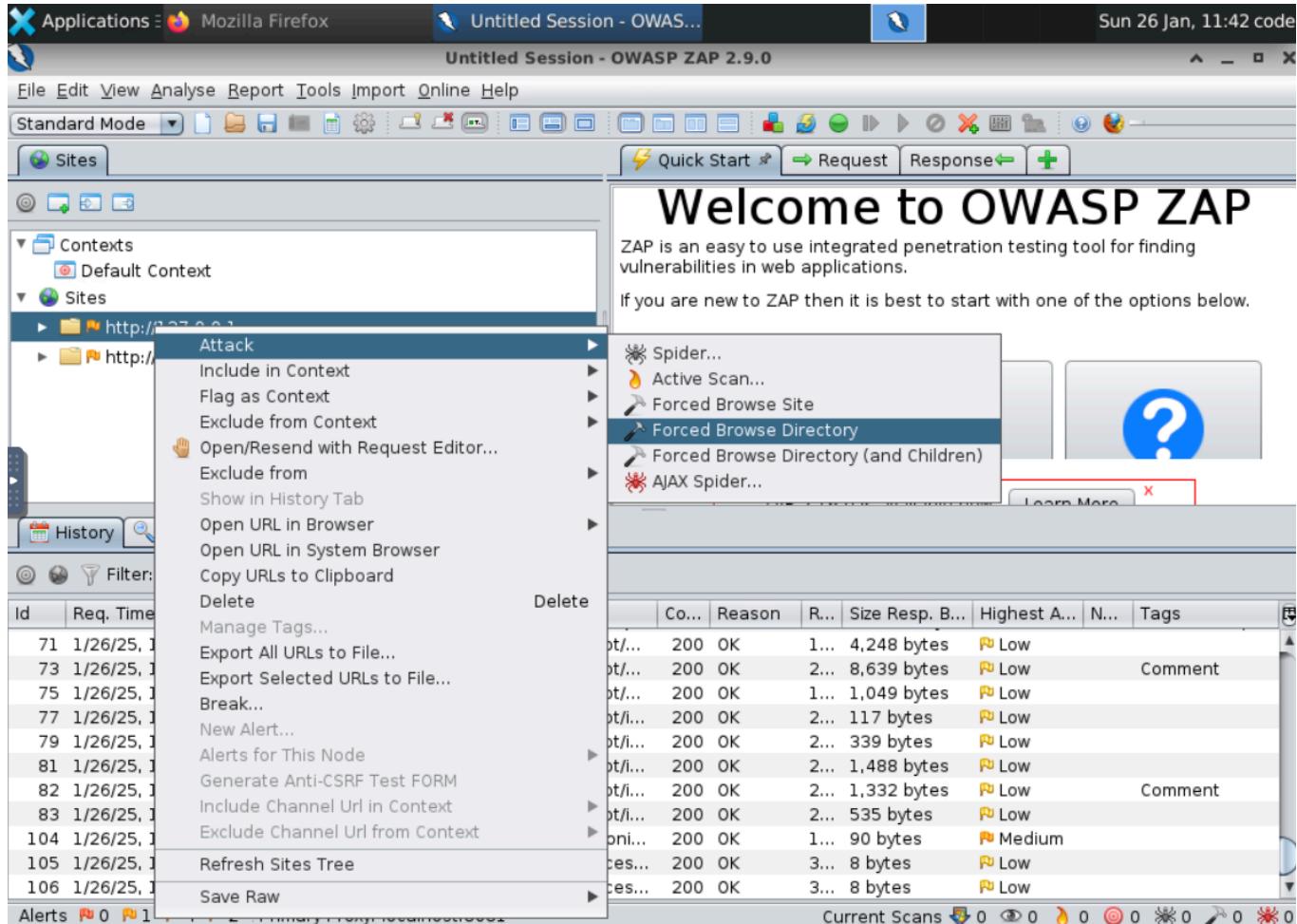
Finding files and folders of a web server using dictionary list

ZAP uses dictionary list of brute-force directories in a web app it tries to access all the directories and files listed used for attacking web authentications and discovering hidden web pages within a web application we can provide the Dic file or list that zap need to use



In options You can find the forced Browse option and then select the dictionary file which you wanted to use and check whether the froxy proxy is in on or not if not on it and enter the url you

wanted to so the forced attack



Using OWASP ZAP TO Spider Crawl a website to Find URL's and Links

Discovering how to use ZAP to crawl or spider a website to find URLs and Links

Web crawlers and spiders can be used to browse every link included in every web page included in the scope requested and keeps a record of every file displayed by it Zap can be used to automate and accelerate this task

ZAP spotter follows the formed responses redirects the URLs within the robots.txt and sitemap.xml files of a web server we can then save those results and analyze them offline This stores all the requests and responses for later analysis

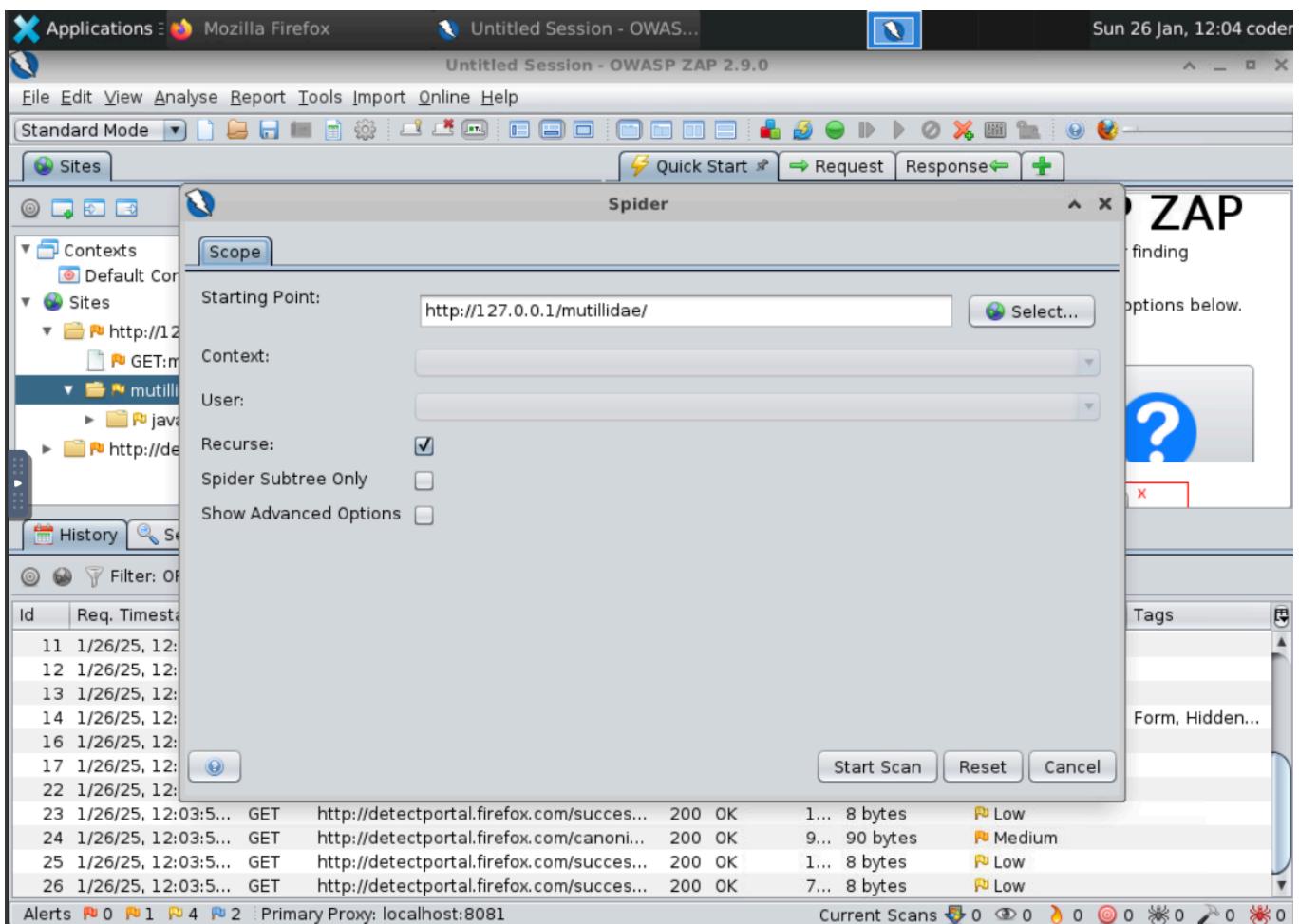
Downloading a site in this way to directory in pc can give us a static copy of the information

Others ways to download like that the static webpages is using wget and httrack but does not save the actual req and responses states of the server

With the spider integrated within OWASP ZAP it will record all the information

The screenshot shows the OWASP ZAP 2.9.0 interface. The left sidebar displays a tree structure of contexts and sites. A context named 'http://127.0.0.1' is selected, and a context menu is open over it. The menu includes options like 'Attack', 'Include in Context', 'Flag as Context', 'Exclude from Context', and 'Open/Resend with Request Editor...'. The main workspace shows a 'Learn More' page with a large 'Learn More' button and a link to a user guide. Below this, there is a table of scan results with columns for ID, Request ID, Date, Response Status, Reason, Response Body Size, Highest Alert Level, and Tags. The table lists several entries, mostly with 'OK' status and low alert levels.

ID	Req. ID	Date	Reason	Size	Resp. B...	Highest A...	N...	Tags
14	1/26/2024	200 OK	6...	55,546 bytes	Medium	Form, Hidden...		
16	1/26/2024	200 OK	8...	90 bytes	Medium			
17	1/26/2024	script/j...	2...	9,845 bytes	Low			
22	1/26/2024	succes...	9...	8 bytes	Low			
23	1/26/2024	succes...	200 OK	1...	8 bytes	Low		
24	1/26/2024	canoni...	200 OK	9...	90 bytes	Medium		
25	1/26/2024	succes...	200 OK	1...	8 bytes	Low		
26	1/26/2024	succes...	200 OK	7...	8 bytes	Low		
27	1/26/2024	canoni...	200 OK	1...	90 bytes	Medium		
28	1/26/2024	succes...	200 OK	2...	8 bytes	Low		
29	1/26/2024	succes...	200 OK	2...	8 bytes	Low		



By scanning You can analyze the results and requests and responses

The screenshot shows the OWASP ZAP 2.9.0 interface. At the top, there are tabs for 'Mozilla Firefox' and 'Untitled Session - OWAS...'. The main window title is 'Untitled Session - OWASP ZAP 2.9.0'. The date 'Sun 26 Jan, 12:05 code...' is displayed at the top right. The menu bar includes 'File', 'Edit', 'View', 'Analyse', 'Report', 'Tools', 'Import', 'Online Help', and 'Standard Mode'. Below the menu is a toolbar with icons for various functions. The left sidebar shows a tree view of scanned files, including 'javascript', 'GET:set-up-database.php', 'styles', 'webservices', 'GET:mutilidae(page)', 'GET:robots.txt', 'GET:sitemap.xml', 'GET:webservices', and 'http://detectportal.firefox.com'. A specific request 'GET:robots.txt' is selected, shown in the center panel with its headers and body. The headers are: HTTP/1.1 404 Not Found, Date: Sun, 26 Jan 2025 12:04:45 GMT, Server: Apache/2.4.52 (Ubuntu), Content-Length: 271, Content-Type: text/html; charset=iso-8859-1. The body contains the HTML response: <!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN"><html><head><title>404 Not Found</title></head><body></body></html>. Below the main panel, there are tabs for 'History', 'Search', 'Alerts', 'Output', 'Spider', and a '+' button. The bottom status bar shows 'New Scan : Progress: 0: http://127.0.0.1/mutilidae' and 'Current Scans: 0 : URLs Found: 496 : Nodes Added: 61 : Export'. There are also buttons for 'Alerts' (0), 'Primary Proxy: localhost:8081', and 'Current Scans'.

We can use this for pentesting like repeating the request that modified data and we can perform active and passive vulnerability scans

Using OWASP ZAP for Viewing and Altering Requests

ZAP is very flexible for modifying the requests, it can change the request method from GET to POST and save the req-response pair to be processed by other tools

For example let us use this tool to modify a valid req and make it invalid with a little bit of malicious SQL injection and then send to server and provoke unexpected behavior from it basically bypassing a client validation

Sun 26 Jan, 13:01 code

Applications Mozilla Firefox Untitled Session - OWAS... 127.0.0.1/mutillidae/ +

127.0.0.1/mutillidae/

⚠ Some of Firefox's security features may offer less protection on your current operating system. [How to fix this issue](#)

Don't show again

OWASP Mutillidae II: Keep Calm and Pwn On

Version: 2.11.15 Security Level: 0 (Hosed) Hints: Enabled Not Logged In

Home | Login/Register | Toggle Hints | Toggle Security | Enforce TLS | Reset DB | View Log | View Captured Data

OWASP 2017	A1 - Injection (SQL)	SQLI - Extract Data	User Info (SQL)
OWASP 2013	A1 - Injection (Other)	SQLI - Bypass Authentication	
OWASP 2010	A2 - Broken Authentication and Session Management	SQLI - Insert Injection	
OWASP 2007	A3 - Sensitive Data Exposure	Blind SQL via Timing	
Web Services	A4 - XML External Entities	SQLMAP Practice	
Others	A5 - Broken Access Control	Via JavaScript Object Notation (JSON)	
Labs	A6 - Security Misconfiguration	Via SOAP Web Service	
Documentation	A7 - Cross Site Scripting (XSS)	Via REST Web Service	
Resources	A8 - Insecure Deserialization		
Donate Today!	A9 - Using Components with Known Vulnerabilities		
Want to Help?			

TIP: Click Hint and Videos on each page

127.0.0.1/mutillidae/index.php?page=user-info.php and Latest Version

We are having this mutillidae which is a built in vulnerability application for practicing web pentesting

Sun 26 Jan, 13:03 co

Applications Mozilla Firefox Untitled Session - OWAS... Sun 26 Jan, 13:03 co

127.0.0.1/mutillidae/index.php + 127.0.0.1/mutillidae/index.php?page=user-info.php

Some of Firefox's security features may offer less protection on your current operating system. [How to fix this issue](#)

User Lookup (SQL)

Back Help Me!

Hints and Videos

Switch to SOAP Web Service version Switch to XPath version

Please enter username and password to view account details

Name
Password

View Account Details

Dont have an account? [Please register here](#)

Donate Want to Help? Video Tutorials

Want to Help?

Video Tutorials

A blue bird icon

Sun 26 Jan, 13:05 co

Applications Mozilla Firefox Untitled Session - OWAS... Sun 26 Jan, 13:05 co

127.0.0.1/mutillidae/index.php + 127.0.0.1/mutillidae/index.php?popUpNotificationCode=SL1&page=

Some of Firefox's security features may offer less protection on your current operating system. [How to fix this issue](#)

OWASP Mutillidae II: Keep Calm and Pwn On

Status Update

Version: 2.11.15 Security Level: 1 (Client-Side Security) [Hints](#) Security level set to 1. Try Slightly Harder.

Home | Login/Register | Toggle Security | Enforce TLS | Reset DB | View Log | View Captured Data

User Lookup (SQL)

Back Help Me!

Hints and Videos

Switch to SOAP Web Service version Switch to XPath version

Please enter username and password to view account details

Raising the security level and testing the sql injection

The screenshot shows a Firefox browser window with the address bar displaying '127.0.0.1/mutillidae/index.php?popUpNotificationCode=SL1&pa'. A security warning message from Firefox is visible at the top: 'Some of Firefox's security features may offer less protection on your current operating system. [How to fix this issue](#)'.

The main content area shows a web page with a 'AJAX' logo and links to 'SWITCH TO SOAP WEB SERVICE VERSION' and 'XML SWITCH TO XPATH VERSION'. A central box prompts the user to 'Please enter username and password to view account details'. Below this, there is a form field labeled 'Name' with the value 'user'. A modal dialog box is overlaid on the page, containing the following text:

- Icon: 127.0.0.1
- Dangerous characters detected. We can't allow these. This all powerful blacklist will stop such attempts.
- Much like padlocks, filtering cannot be defeated.
- Blacklisting is I33t like I33tspeak.

An 'OK' button is located at the bottom right of the dialog.

We can see the popup which is because of the client side validation to bypass this we need to break the request from going to the server by ZAP and change the Request we cannot see the req in our zap so we need to to bypass

To do this we need to create a valid req and then intercept it with the ZAP proxy and we need to enable the request interception breakpoint within the ZAP



We can see a green button but hover over it it changes to red which is for setting up break on all req and responses

The screenshot shows the OWASP ZAP 2.9.0 interface. The top navigation bar includes 'Applications' (with a blue icon), 'Mozilla Firefox' (with a red icon), 'Untitled Session - OWAS...' (with a blue icon), and 'Sun 26 Jan, 13:31 coder' (with a yellow icon). The main window has a toolbar with icons for file operations, search, and analysis. A left sidebar lists 'Contexts' (Default Context) and 'Sites' (http://r10.o.lencr.org, http://127.0.0.1, http://detectportal.firefox.com). The central area shows a request configuration panel with 'Method: GET', 'Header: Text', and 'Body: Text'. The body contains a GET request to http://127.0.0.1/mutillidae/index.php?page=user-info.php&username=user&password=password&user-info-php-submit-button=View+Account+Details. Below this is a large text input field. At the bottom, there are tabs for 'History', 'Search', 'Alerts', 'Output', and a green '+' button. The main table displays captured traffic with columns: d, Req. Timestamp, Meth..., URL, Co..., Reason, R..., Size, Resp. B..., Highest A..., N..., and Tags. The table rows show various GET requests from 1/26/25 at different times to the detectportal.firefox.com domain, with responses ranging from 200 OK to 302 Found, sizes from 8 bytes to 90 bytes, and severity levels from Low to Medium.

d	Req. Timestamp	Meth...	URL	Co...	Reason	R...	Size	Resp. B...	Highest A...	N...	Tags
45	1/26/25, 1:10:2...	GET	http://detectportal.firefox.com/succes...	200	OK	3...	8 bytes		Low		
46	1/26/25, 1:10:2...	GET	http://detectportal.firefox.com/succes...	200	OK	3...	8 bytes		Low		
47	1/26/25, 1:15:1...	GET	http://detectportal.firefox.com/canoni...	200	OK	1...	90 bytes		Medium		
48	1/26/25, 1:15:1...	GET	http://detectportal.firefox.com/succes...	200	OK	2...	8 bytes		Low		
49	1/26/25, 1:15:1...	GET	http://detectportal.firefox.com/succes...	200	OK	2...	8 bytes		Low		
50	1/26/25, 1:17:0...	GET	http://detectportal.firefox.com/canoni...	200	OK	1...	90 bytes		Medium		
51	1/26/25, 1:17:0...	GET	http://detectportal.firefox.com/succes...	200	OK	3...	8 bytes		Low		
52	1/26/25, 1:17:0...	GET	http://detectportal.firefox.com/succes...	200	OK	3...	8 bytes		Low		
53	1/26/25, 1:18:2...	GET	http://detectportal.firefox.com/canoni...	200	OK	2...	90 bytes		Medium		
54	1/26/25, 1:18:2...	GET	http://detectportal.firefox.com/succes...	200	OK	3...	8 bytes		Low		
55	1/26/25, 1:18:2...	GET	http://detectportal.firefox.com/succes...	200	OK	3...	8 bytes		Low		

By pressing the play back button the req sends to the server

The screenshot shows a Firefox browser window with three tabs: 'Applications', 'Mozilla Firefox', and 'Untitled Session - OWAS...'. The main content area displays a web page from '127.0.0.1/mutillidae/index.php?page=user-info.php&username=user'. A security warning message at the top left states: 'Some of Firefox's security features may offer less protection on your current operating system. [How to fix this issue](#)'.

The page itself has a sidebar on the left with links like 'Web Services', 'Others', 'Labs', 'Documentation', 'Resources', 'Donate', 'Want to Help?', 'Video Tutorials', 'Announcements', and 'Getting Started'. The main content area includes a 'AJAX' logo, two buttons for 'Switch to SOAP Web Service version' and 'Switch to XPath version', and a red box containing the text 'Authentication Error: Bad user name or password'. Below this is a brown box with the instruction 'Please enter username and password to view account details'. There are input fields for 'Name' and 'Password', and a 'View Account Details' button. At the bottom, there is a link 'Dont have an account? [Please register here](#)' and a message 'Results for "user".0 records found.'

we can see a authentication error stating a bad username or password

Now again enter the valid name and password and do not press for view account details
Go to the ZAP and set the breakpoint and then send the req by pressing the view account details

The screenshot shows the OWASP ZAP 2.9.0 interface. The top bar displays 'Applications' and 'Mozilla Firefox' tabs, the date 'Sun 26 Jan, 13:35 code', and a toolbar with various icons. The main window has a 'Sites' tab selected. On the left, a tree view shows 'Contexts' (Default Context) and 'Sites' (http://r10.o.lencr.org, http://127.0.0.1, http://detectportal.firefox.com). The right panel shows a 'Request' tab with a method 'GET' to 'http://127.0.0.1/mutillidae/index.php?page=user-info.php&username=user&password=password+&user-info-php-submit-button=View+Account+Details'. The bottom section shows a table of network requests with columns: Id, Req. Timestamp, Meth..., URL, Co..., Reason, R..., Size, Resp. B..., Highest A..., N..., Tags. The table lists 12 requests from 1/26/25, 1:17:0... to 1/26/25, 1:34:2... to the same URL, with varying response codes (200 OK) and sizes (8 bytes to 56,938 bytes). A 'Tags' column indicates 'Form, Passwo...' for the first request.

ID	Req. Timestamp	Meth...	URL	Co...	Reason	R...	Size	Resp. B...	Highest A...	N...	Tags
52	1/26/25, 1:17:0...	GET	http://detectportal.firefox.com/succes...	200	OK	3...	8 bytes		Low		
53	1/26/25, 1:18:2...	GET	http://detectportal.firefox.com/canoni...	200	OK	2...	90 bytes		Medium		
54	1/26/25, 1:18:2...	GET	http://detectportal.firefox.com/succes...	200	OK	3...	8 bytes		Low		
55	1/26/25, 1:18:2...	GET	http://detectportal.firefox.com/succes...	200	OK	3...	8 bytes		Low		
56	1/26/25, 1:31:5...	GET	http://127.0.0.1/mutillidae/index.php?...	200	OK	6...	56,938 bytes		Medium		Form, Passwo...
57	1/26/25, 1:32:5...	GET	http://detectportal.firefox.com/canoni...	200	OK	2...	90 bytes		Medium		
58	1/26/25, 1:32:5...	GET	http://detectportal.firefox.com/succes...	200	OK	3...	8 bytes		Low		
59	1/26/25, 1:32:5...	GET	http://detectportal.firefox.com/succes...	200	OK	3...	8 bytes		Low		
60	1/26/25, 1:34:2...	GET	http://detectportal.firefox.com/canoni...	200	OK	2...	90 bytes		Medium		
61	1/26/25, 1:34:2...	GET	http://detectportal.firefox.com/succes...	200	OK	2...	8 bytes		Low		
62	1/26/25, 1:34:2...	GET	http://detectportal.firefox.com/succes...	200	OK	2...	8 bytes		Low		

Here we have intercepted the request which is bypassed the client validation by the application but have not sent to the server because we have stopped it

now unsetting the breakpoint and playing on the req and return with injecting special characters

The screenshot shows a Firefox browser window with the URL `127.0.0.1/mutillidae/index.php?page=user-info.php&username=user`. A warning message at the top states: "Some of Firefox's security features may offer less protection on your current operating system. [How to fix this issue](#)". Below it, a "Don't show again" button is visible. On the left, there are sidebar links for "Video Tutorials", "Announcements" (with a blue bird icon), and "Getting Started". The main content area displays an "Error Message" titled "Failure is always an option". The message details a SQL syntax error:

Line	238
Code	0
File	/var/www/html/mutillidae/classes/MySQLHandler.php
Message	/var/www/html/mutillidae/classes/MySQLHandler.php on line 230: 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 'password' '' at line 2 Query: SELECT * FROM accounts WHERE username='user' AND password='password' '(1064) [mysql_sql_exception]
Trace	#0 /var/www/html/mutillidae/classes/MySQLHandler.php(328): MySQLHandler->doExecuteQuery() #1 /var/www/html/mutillidae/classes/SQLQueryHandler.php(356): MySQLHandler->executeQuery() #2 /var/www/html/mutillidae/user-info.php(171): SQLQueryHandler->getUserAccount() #3 /var/www/html/mutillidae/index.php(513): require_once('...') #4 {main}
Diagnostic Information	Error attempting to display user information

A red button at the bottom says "Click here to reset the DB". At the bottom of the page, the browser and PHP versions are listed: "Browser: Mozilla/5.0 (X11; Linux x86_64; rv:134.0) Gecko/20100101 Firefox/134.0" and "PHP Version: 8.1.2-lubuntu2.18".

we can see a new error this is because the server was unable to process the input correctly but we bypassed the client validation which stopped earlier however this is also a client validation but we bypassed some of the client validation

we can say that the web app cannot process the unexpected server side request correctly

By Using this we can understand how the web application works

Conclusion

OWASP ZAP a web security testing tool has various benefits not only the tasks done by me This tool is a Great tool to do penetration testing and reporting the web applications from intercepting the traffic and finding the links and URLs of the target web application and a spider to crawl the website and finding the directories and altering the requests and analyzing the responses and checking for vulnerabilities