VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

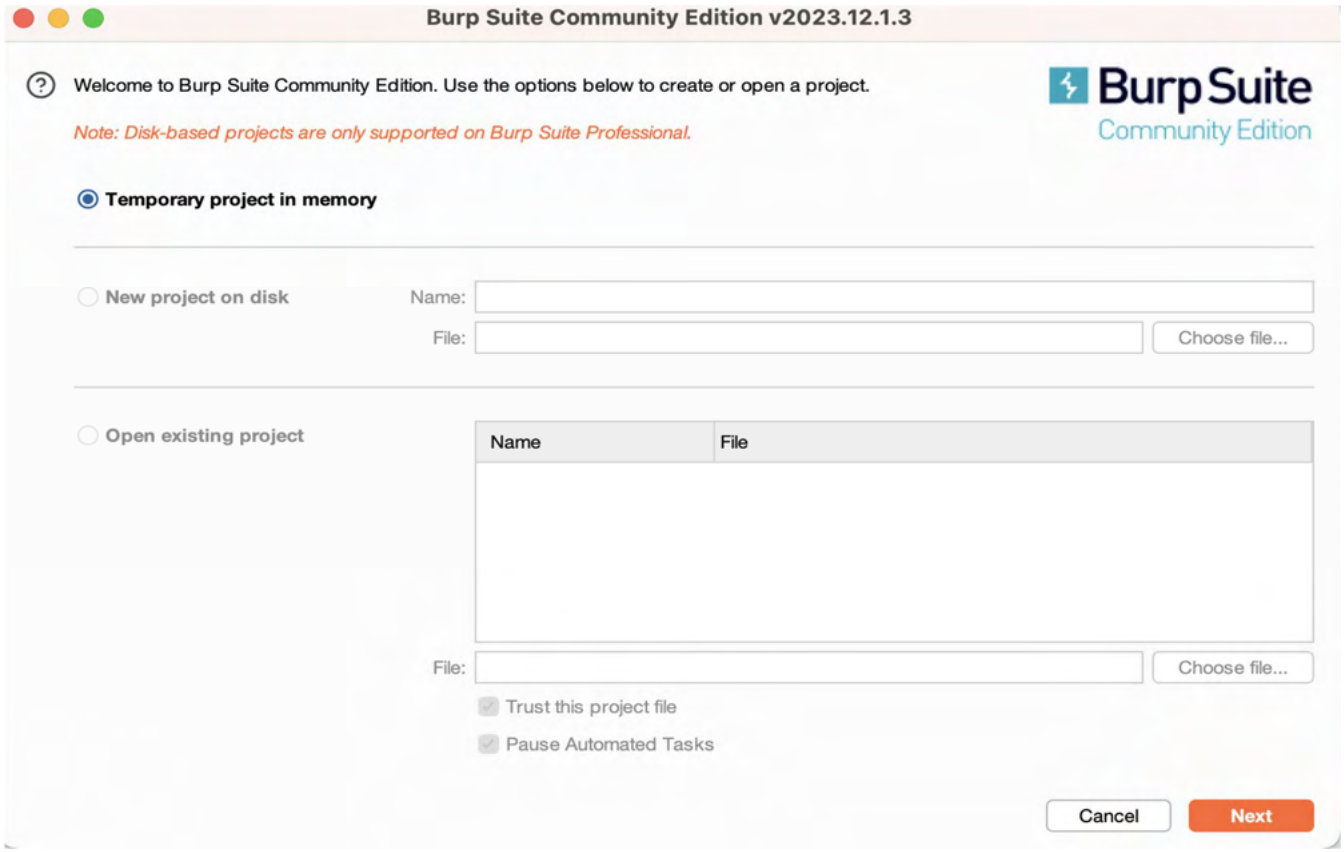
INFORMATION SECURITY MANAGEMENT LAB

EXPERIMENT-3

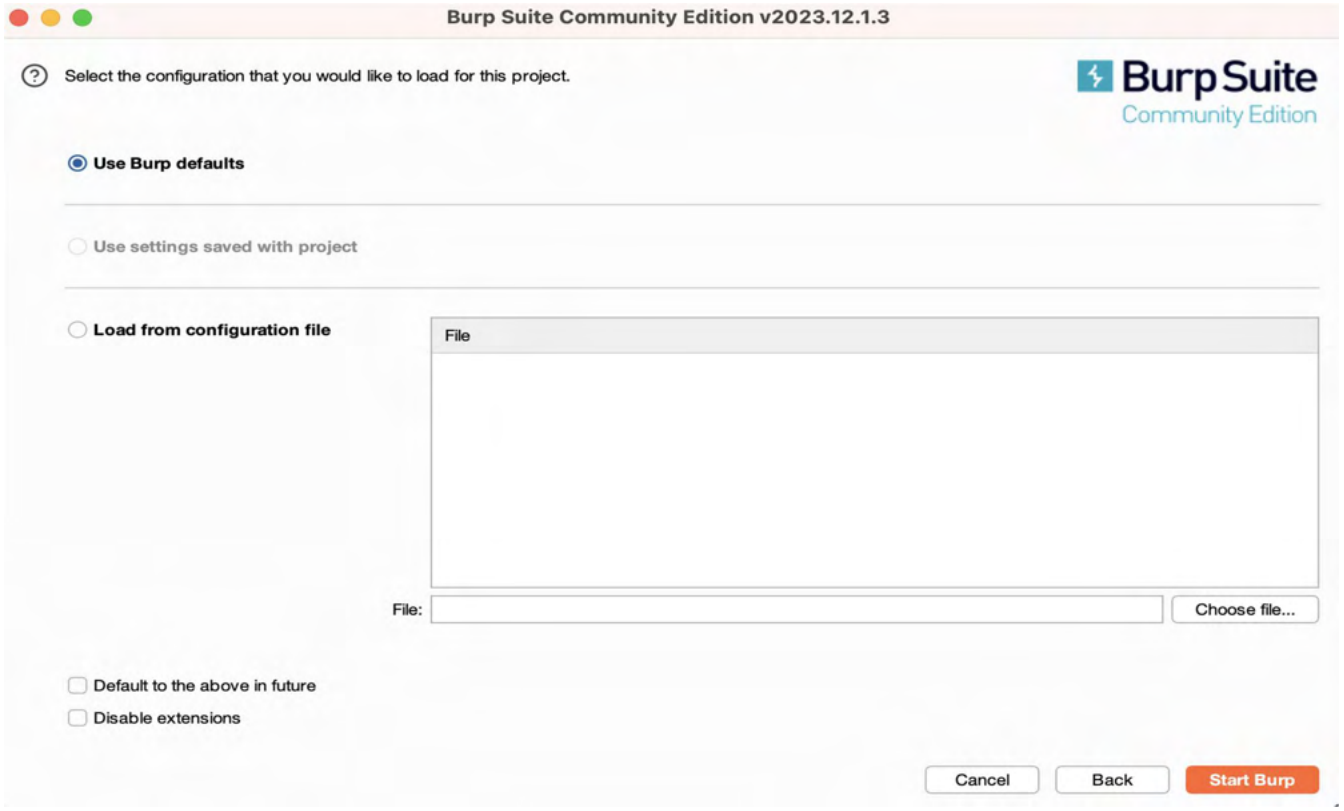
Features and Functionalities of Burp Suite

GROUP NO. :	11
TEAM MEMBER 1 :	Namit Mehrotra
REG. NO. :	21BCE0763
TEAM MEMBER 2 :	Purva Sharma
REG.NO :	21BCE0169
SUBJECT CODE :	BCSE354E
SUBJECT TITLE :	Information Security Management
LAB SLOT :	L29+L30
SEMESTER :	Winter Semester 2023-2024
GUIDED BY :	NIHA K

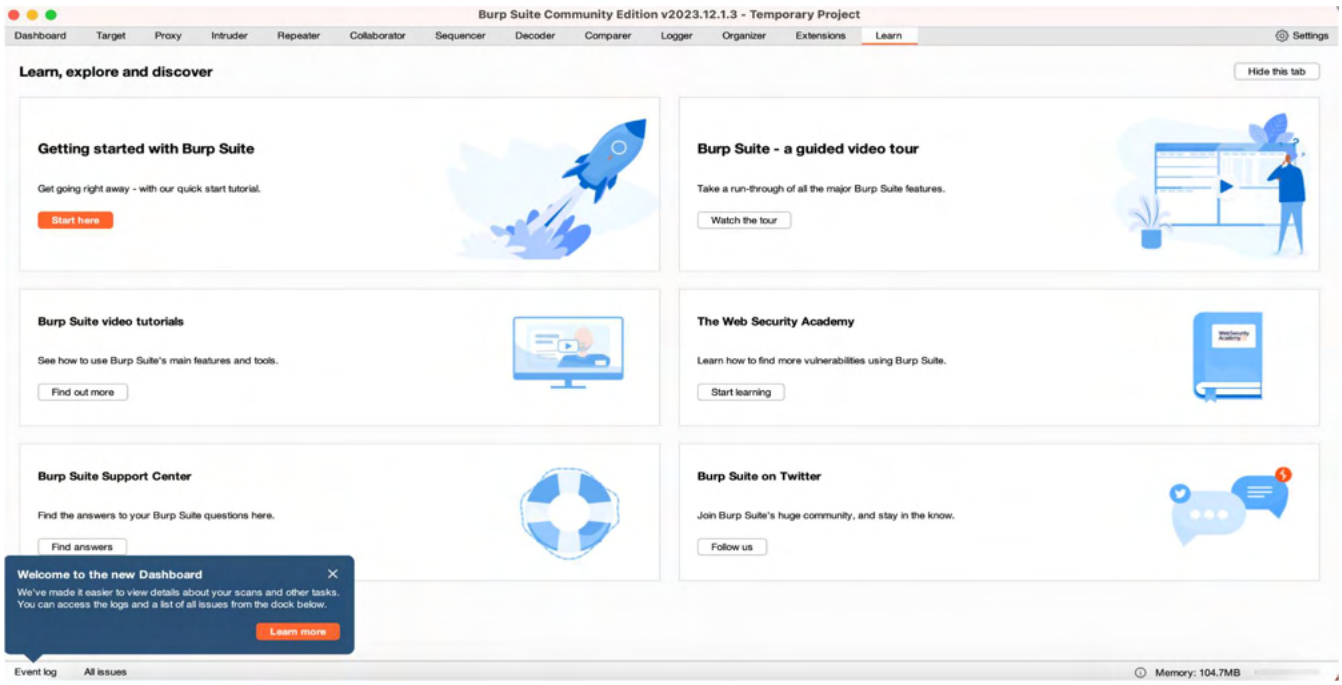
Features and Functionalities of Burp Suite:



-> Next



-> Start Burp



Manual penetration testing features:

- **Intercept everything your browser sees:** Burp Suite's built-in browser works right out of the box - enabling you to modify every HTTP message that passes through it.
- **Quickly assess your target:** Determine the size of your target application. Auto-enumeration of static and dynamic URLs, and URL parameters.
- **Speed up granular workflows:** Modify and reissue individual HTTP and WebSocket messages, and analyze the response - within a single window.
- **Manage recon data:** All target data is aggregated and stored in a target site map - with filtering and annotation functions.
- **Expose hidden attack surface:** Find hidden target functionality with an advanced automatic discovery function for "invisible" content.
- **Break HTTPS effectively:** Proxy even secure HTTPS traffic, using Burp Suite's built-in instrumented browser.
- **Work with HTTP/2:** Burp Suite offers unrivalled support for HTTP/2-based testing - enabling you to work with HTTP/2 requests in ways that other tools cannot.
- **Work with WebSockets:** WebSockets messages get their own specific history - allowing you to view and modify them.
- **Manually test for out-of-band vulnerabilities:** Make use of a dedicated client to incorporate Burp Suite's out-of-band (OAST) capabilities during manual testing.
- **DOM Invader:** Use Burp Suite's built-in browser to test for DOM XSS vulnerabilities more easily - with DOM Invader.
- **Assess token strength:** Easily test the quality of randomness in data items intended to be unpredictable (e.g. tokens).

Advanced / custom automated attacks:

- **Faster brute-forcing and fuzzing:** Deploy custom sequences of HTTP requests containing multiple payload sets. Radically reduce time spent on many tasks.
- **Query automated attack results:** Capture automated results in customized tables, then filter and annotate to find interesting entries / improve subsequent attacks.
- **Construct CSRF exploits:** Easily generate CSRF proof-of-concept attacks. Select any suitable request to generate exploit HTML.
- **Facilitate deeper manual testing:** See reflected/stored inputs even when a bug is not confirmed. Facilitates testing for issues like XSS.
- **Scan as you browse:** The option to passively scan every request you make, or to perform active scans on specific URLs.
- **Automatically modify HTTP messages:** Settings to automatically modify responses. Match and replace rules for both responses and requests.

Automated scanning for vulnerabilities:

- **Browser-powered scanning:** Burp Scanner uses its embedded browser to render its target - enabling it to navigate even complex single-page applications (SPAs).
- **Harness pioneering OAST technology:** High signal: low noise. Scan with pioneering, friction-free, out-of-band-application security testing (OAST).
- **Remediate bugs effectively:** Custom descriptions and step-by-step remediation advice for every bug, from PortSwigger Research and the Web Security Academy.
- **Fuel vulnerability coverage with research:** Cutting-edge scan logic from PortSwigger Research combines with coverage of over 100 generic bugs.
- **BChecks:** Create custom scan checks for Burp Scanner, written in a simple text-based language.
- **API scanning:** Discover more potential attack surfaces. Burp Scanner parses JSON or YAML API definitions - scanning any API endpoints it finds.
- **Authenticated scanning:** Scan privileged areas of target applications, even if they use complex login mechanisms like single sign-on (SSO).
- **Conquer client-side attack surfaces:** A built-in JavaScript analysis engine helps to find holes in client-side attack surfaces.
- **Configure scan behavior:** Customize what you audit, and how. Skip specific checks, fine-tune insertion points, and much more. Or use preset scan modes to get an overview.

Productivity tools:

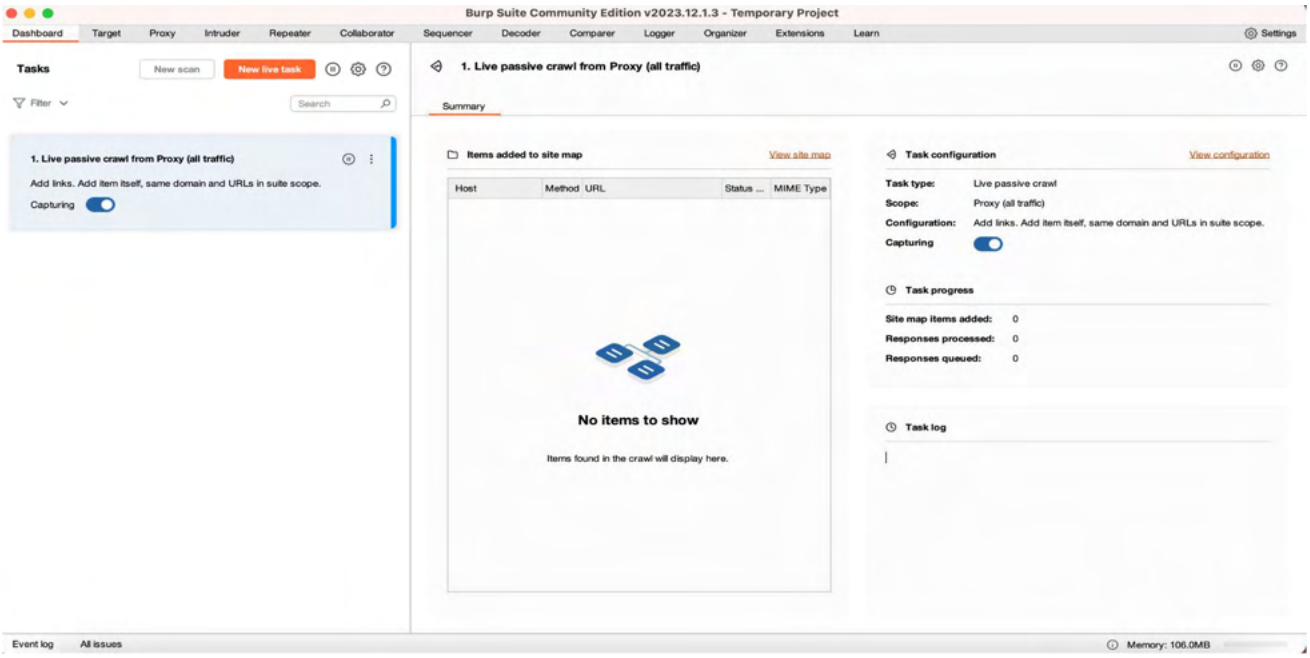
- **Deep-dive message analysis:** Show follow-up, analysis, reference, discovery, and remediation in a feature-rich HTTP editor.
- **Utilize both built-in and custom configurations:** Access predefined configurations for common tasks, or save and reuse custom configurations.
- **Project files:** Auto-save everything you do while on an engagement, as well as the configuration settings you use.
- **Burp Logger:** See every HTTP message that passes through Burp Suite's tools - all in one place - with Burp Logger.
- **Speed up data transformation:** Decode or encode data, with multiple built-in operations (e.g. Hex, Octal, Base64).
- **Burp Organizer:** Store and annotate interesting messages you find while testing, so you can come back to them later.

- **Make code more readable:** Automatically pretty-print code formats including JSON, JavaScript, CSS, HTML, and XML.
- **Easily remediate scan results:** See source, discovery, contents, and remediation, for every bug, with aggregated application data.
- **Search function:** Search everywhere in Burp Suite Professional at once, with its powerful search function.
- **Simplify scan reporting:** Customize with HTML / XML formats. Report all evidence identified, including issue details.

<u>BApp extensions:</u>

- **Create custom extensions:** The Montoya API ensures universal adaptability. Code custom extensions to make Burp work for you.
- **Hackvertor:** Convert between various encodings with Hackvertor. Use multiple nested tags to perform layered encoding. Even execute your own code with custom tags - and more.
- **Autorize:** When testing for authorization vulnerabilities, save time and perform repeat requests with Autorize.
- **Turbo Intruder:** Configured in Python, with a custom HTTP stack, Turbo Intruder can unleash thousands of requests per second.
- **J2EE Scan:** Expand your Java-specific vulnerability catalogue and hunt the most niche bugs, with J2EEScan.
- **Access the extension library:** The BApp Store customizes and extends capabilities. Over 250 extensions, written and tested by Burp users.
- **Upload Scanner:** Adapt Burp Scanner's attacks by uploading and testing multiple file-type payloads, with Upload Scanner.
- **HTTP Request Smuggler:** Scan for request smuggling vulnerabilities - and exploit them more easily by having HTTP Request Smuggler tweak offsets automatically for you.
- **Param Miner:** Quickly find unkeyed inputs with Param Miner - can guess up to 65,000 parameter names per second.
- **Backslash Powered Scanner:** Find research-grade bugs, and bridge human intuition and automation, with Backslash Powered Scanner.

Dashboard:



The Burp Suite dashboard is a central hub for managing and controlling various aspects of the tool. It consists of several tabs, each serving a specific purpose. Let's dive into each section in detail:

1. <u>Target Tab</u>: The "Target" tab in Burp Suite is a crucial component that allows users to manage and control the scope of their testing.

Functionality:

- Identifies and manages the target scope for testing.
- Allows you to add, remove, or modify target scope.

How to Use:

- Add a target by entering the URL and clicking "Add to Scope."
- Modify scope options like including or excluding specific URLs or entire domains.

2. <u>Proxy Tab</u>: The Proxy tab in Burp Suite is a powerful tool that allows you to intercept and manipulate HTTP/S traffic between your browser and the target web application.

Functionality:

- Manages proxy settings for intercepting and modifying HTTP/S traffic.
- Shows intercepted requests for analysis.

How to Use:

- Configure browser proxy settings to use Burp.
- Intercept requests, modify them, and forward them to the server.

3. <u>Spider Tab</u>: Automatically crawls the target web application in Burp Suite, mapping its structure to identify accessible content and functionality. Set the scope in the Target tab, initiate the Spider, and analyze results in the "Spider" sub-tab for discovered URLs.

Functionality:

- Crawls the target application, discovering and mapping its structure.
- Helps identify all accessible content and functionality.

How to Use:

- Set the scope in the Target tab and click "Spider."
- Analyze results in the "Spider" sub-tab for discovered URLs.

4. <u>Scanner Tab</u>: Automates security testing by scanning the target for vulnerabilities. Configure settings, launch the scanner to identify and report potential security issues in the web application.

Functionality:

- Automates the identification of security vulnerabilities.
- Integrates various scanning tools.

How to Use:

- Select the target, configure scan settings, and click "Start scan."
- Review the scan results in the "Scanner" sub-tab.

5. <u>Intruder Tab</u>: Burp Intruder is a powerful tool for performing highly customizable, automated attacks against websites. It enables you to configure attacks that send the same request over and over again, inserting different payloads into predefined positions each time. Among other things, you can use Intruder to:

- Fuzz for input-based vulnerabilities.
- Perform brute-force attacks.
- Enumerate valid identifiers and other inputs.
- Harvest useful data.

Functionality:

- Automates customized attacks on web applications.
- Aids in identifying vulnerabilities through parameter manipulation.

How to Use:

- Define attack positions, payloads, and other settings.
- Launch the attack and analyze the responses in the "Intruder" sub-tab.

6. <u>Repeater Tab:</u> Burp Repeater is a tool that enables you to modify and send an interesting HTTP or WebSocket message over and over.

You can use Repeater for all kinds of purposes, for example to:

- Send a request with varying parameter values to test for input-based vulnerabilities.
- Send a series of HTTP requests in a specific sequence to test for vulnerabilities in multi-step processes, or vulnerabilities that rely on manipulating the connection state.
- Manually verify issues reported by Burp Scanner.

Functionality:

- Allows manual testing and analysis of individual HTTP requests.
- Useful for fine-tuning or retesting specific requests.

How to Use:

- Select a request in the Proxy history and send it to the Repeater.
- Modify and resend the request, and observe responses.

7. <u>Decoder Tab:</u> Burp Decoder enables you to transform data using common encoding and decoding formats. You can use Decoder to:

- Manually decode data.
- Automatically identify and decode recognizable encoding formats, such as URL-encoding.
- Transform raw data into various encoded and hashed formats.

Decoder enables you to apply layers of transformations to the same data. This enables you to unpack or apply complex encoding schemes. For example, to generate modified data in the correct format for an attack, you could:

1. Apply URL-decoding, then HTML-decoding.
2. Edit the decoded data.
3. Reapply the HTML-encoding, then the URL-encoding.

Functionality:

- Decodes and encodes data to facilitate analysis.
- Supports various encoding schemes (Base64, URL, etc.).

How to Use:

- Paste encoded data, select the encoding type, and decode.

8. **Comparer Tab:** Burp Comparer enables you to compare any two items of data. You can use Comparer to quickly and easily identify subtle differences between requests or responses. For example:

- To compare responses to failed logins that use valid and invalid usernames, for username enumeration.
- To compare large responses with different lengths that you have identified in an Intruder attack.
- To compare similar requests that give rise to different application behavior.
- To compare responses when testing for blind SQL injection bugs using Boolean condition injection, to see whether injecting different conditions results in a relevant difference in responses.

Functionality:

- Compares two pieces of data or requests for differences.
- Useful for identifying variations in responses.

How to Use:

- Paste or load two pieces of data, click "Compare," and analyze the differences.

9. **Extender Tab:** Enables customizing and extending Burp Suite's functionality. Use the Extender tab to load and manage extensions, scripts, or plugins for tailored testing and automation.

Functionality:

- Allows the integration of third-party extensions.
- Extends Burp's functionality with custom tools and scripts.

How to Use:

- Manage and load extensions from the BApp Store.
- Create your own extensions to enhance capabilities.

10. <u>Options Tab:</u> Configures global settings in Burp Suite, allowing users to customize preferences, proxy listeners, and other application-wide parameters. Adjust settings for optimal testing and workflow efficiency.

Functionality:

- Configures global settings for Burp Suite.
- Customizes various aspects like display, proxy, and security settings.

How to Use:

- Adjust settings according to your testing requirements.

11. <u>Project Tab</u>: Organizes testing activities within Burp Suite, facilitating the management of multiple projects. Create, save, and load project files, enabling efficient collaboration and organization of scan data and configurations.

Functionality:

- Manages multiple projects for organized testing.
- Saves and loads project configurations.

How to Use:

- Create, save, and load projects for different applications

12. <u>Alerts Tab</u>: Displays and tracks security alerts generated during testing in Burp Suite. View detailed information on identified vulnerabilities, prioritize remediation efforts, and manage the security findings efficiently.

Functionality:

- Lists and categorizes discovered vulnerabilities.
- Provides detailed information on each issue.

How to Use:

- Review alerts after scans to prioritize and address vulnerabilities.

The Burp Suite dashboard is a comprehensive interface that empowers users to perform a wide range of security testing activities, from initial mapping and analysis to automated scanning and manual exploitation. Each tab serves a specific purpose, contributing to the overall effectiveness of the tool in identifying and mitigating security risks in web applications.

➢ Proxy Tab:

The Proxy tab in Burp Suite is a powerful tool that allows you to intercept and manipulate HTTP/S traffic between your browser and the target web application. Here's a detailed overview of the Proxy tab:

Proxy Tab Overview:

Interception:

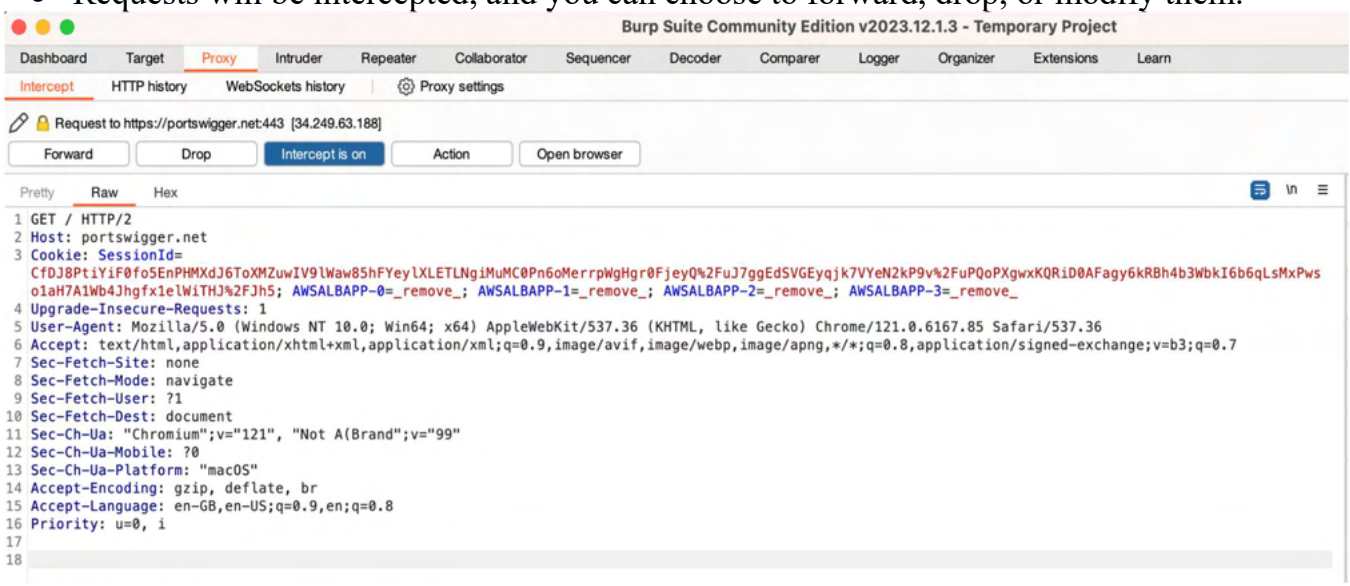
Functionality:

- Intercepts and allows modification of HTTP/S requests and responses.
- Essential for manual testing and analysis of web application traffic.

How to Use:

- Enable the interception by clicking "Intercept is on" in the Proxy tab.

- Requests will be intercepted, and you can choose to forward, drop, or modify them.



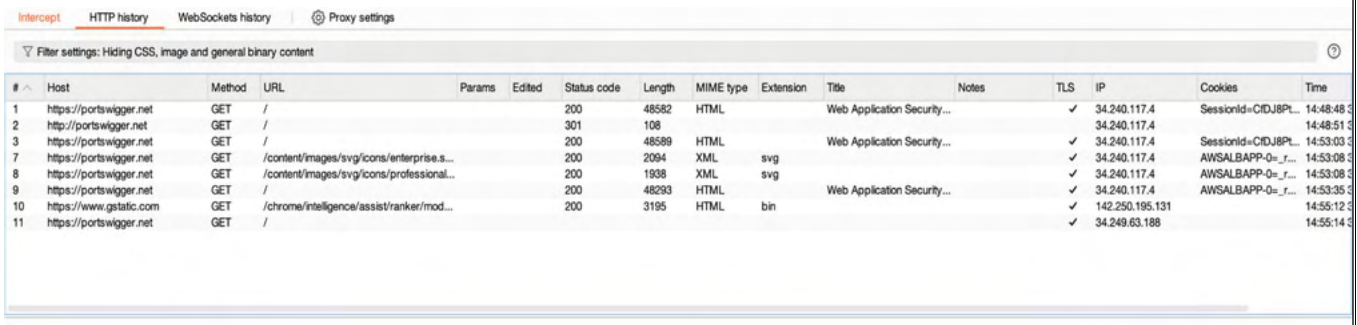
HTTP History:

Functionality:

- Logs all HTTP/S requests and responses passing through the proxy.
- Allows for easy review and analysis of the traffic.

How to Use:

- View and filter the HTTP history to inspect requests and responses.



Scope:

Functionality:

- Defines the target scope for testing.
- Allows you to include or exclude specific URLs or entire domains.

How to Use:

- Configure scope settings in the Target tab to focus testing on specific areas.

Options:

Functionality:

- Configures various proxy options and settings.
- Includes options for interception, request handling, and display.

How to Use:

- Adjust proxy options according to your testing requirements.

WebSockets:

Functionality:

- Handles WebSocket traffic for applications using this communication protocol.
- Enables interception and analysis of WebSocket messages.

How to Use:

- Enable WebSocket support in the Proxy options and monitor WebSocket messages.

HTTP/2:

Functionality:

- Supports the interception and analysis of HTTP/2 traffic.
- Allows for testing and manipulation of applications using HTTP/2.

How to Use:

- Enable HTTP/2 support in the Proxy options and monitor HTTP/2 traffic.

Options Sub-Tab:

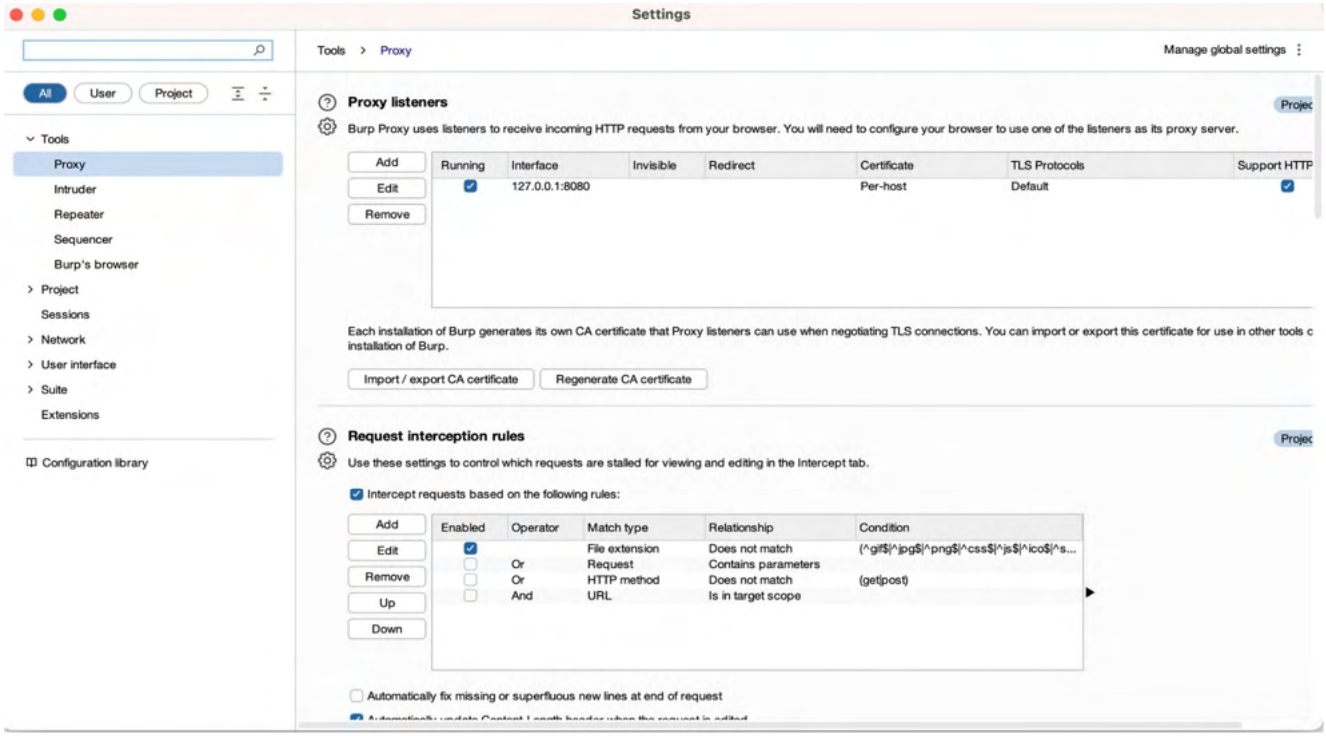
Functionality:

- Provides detailed configuration options for the proxy.
- Allows customization of proxy listeners, interception rules, and more.

How to Use:

- Access the Options sub-tab to fine-tune proxy settings based on your requirements.

Proxy Settings:



How to Set up Burp Suite Proxy:

In Burp Suite, the Certificate Authority (CA) certificate is a crucial component when using the Proxy tool. The Proxy tool allows you to intercept and manipulate HTTP/S traffic between your browser and the target web application. When you enable interception in Burp Suite, it acts as a proxy between your browser and the target server, allowing you to view and modify the requests and responses.

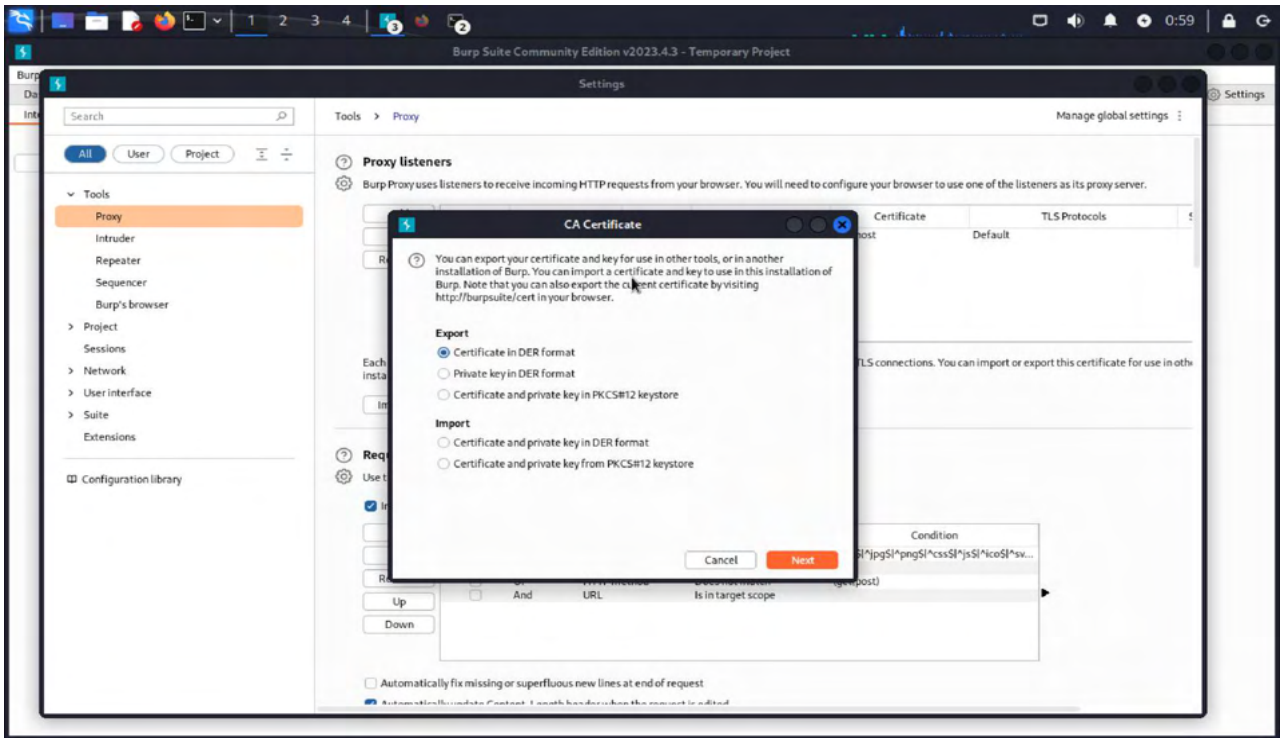
Here's how the CA certificate works in the Proxy tab:

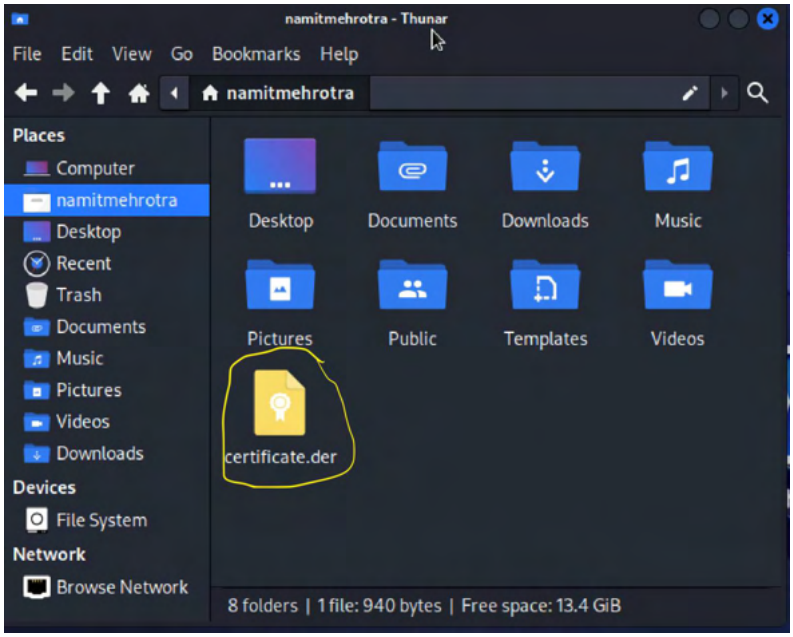
1. **Generate CA Certificate:** Burp Suite generates its own CA certificate, which is used to sign SSL certificates for the sites you visit. This CA certificate is unique to your Burp Suite instance.
2. **Install CA Certificate:** To intercept and modify HTTPS traffic, your browser must trust the Burp Suite CA certificate. You need to install this CA certificate in your browser's certificate store. The CA certificate is usually found in the "User Options" section under the "Proxy" tab in Burp Suite.
3. **Intercept HTTPS Traffic:** Once the CA certificate is installed and the interception is enabled in the Proxy tab, Burp Suite can decrypt and inspect HTTPS traffic between your browser and the target server. This allows you to see and modify the content of encrypted connections.
4. **Configure Browser:** After installing the CA certificate, you need to configure your browser to use Burp Suite as a proxy. Set the browser's proxy settings to use Burp Suite as the proxy server on a specific port (default is 127.0.0.1:8080).
5. **SSL Handshake:** When you visit an HTTPS site, the SSL handshake occurs. Burp Suite generates a new SSL certificate for the target site signed by its CA certificate. Since your

browser trusts the Burp Suite CA, it accepts the certificate, allowing Burp Suite to intercept and modify the encrypted traffic.

6. **Modify Requests and Responses:** With the CA certificate in place, you can now intercept and modify both HTTP and HTTPS traffic using Burp Suite. This is useful for security testing, debugging, and analyzing how applications handle different types of requests and responses.

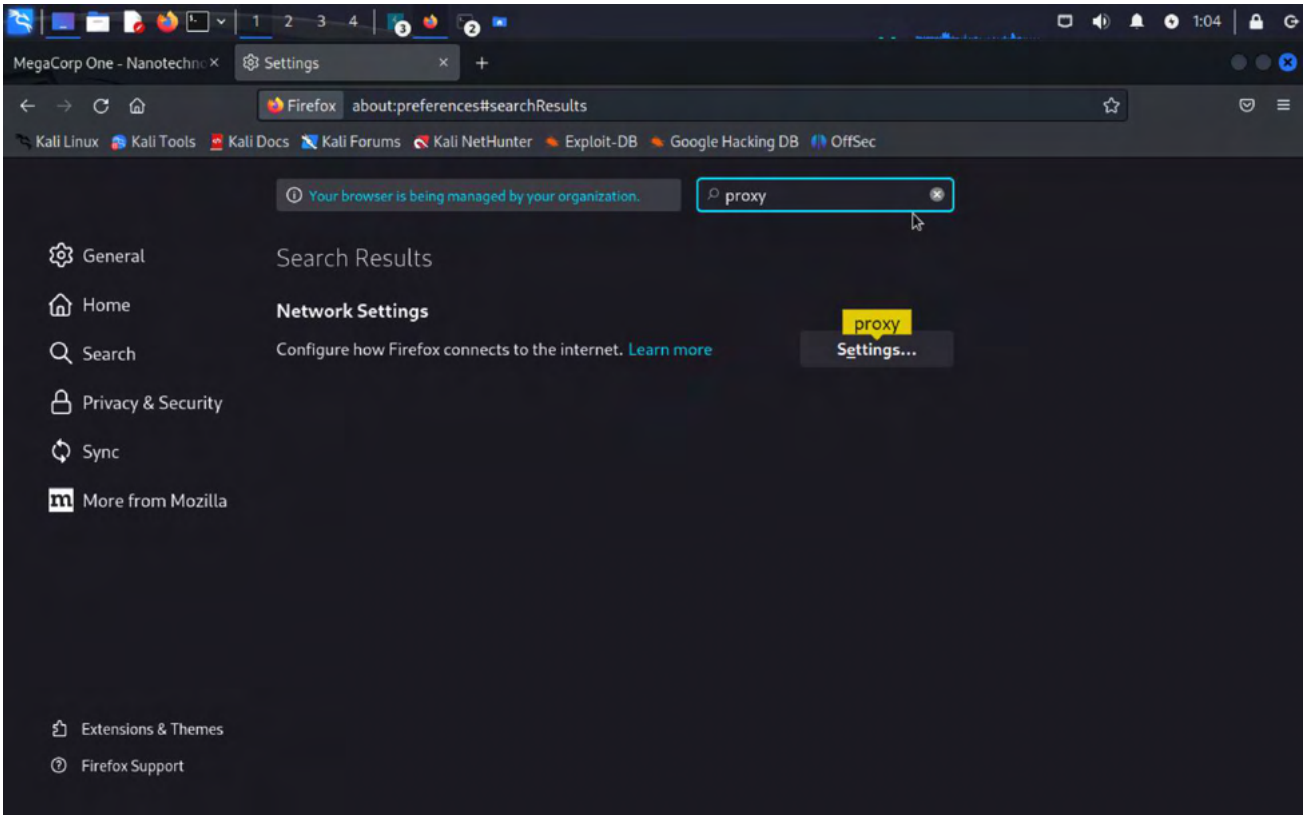
Remember that using a proxy to intercept HTTPS traffic requires careful handling and compliance with ethical and legal standards. It's typically used for security testing and debugging in controlled environments. Always ensure that you have the necessary permissions and adhere to relevant laws and regulations when intercepting and manipulating network traffic.



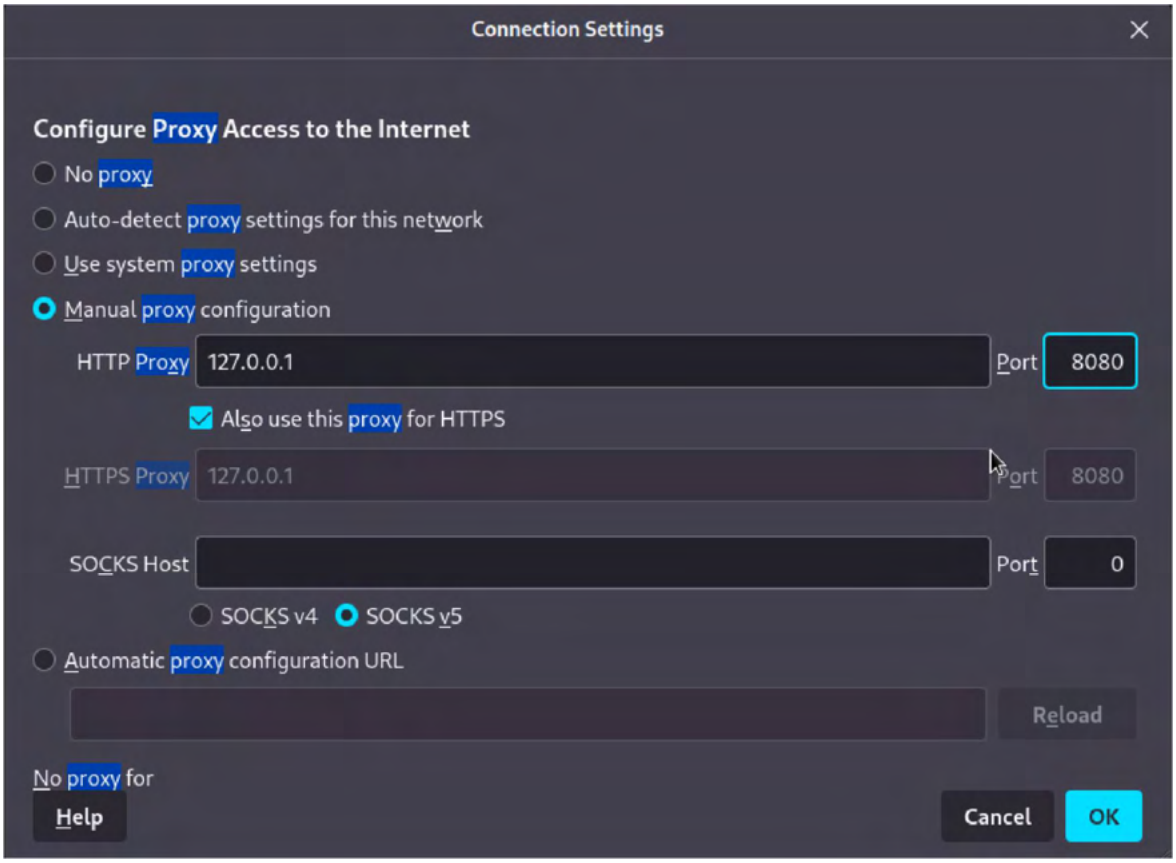


How to Use Burp Suite Proxy:

Configure Browser:

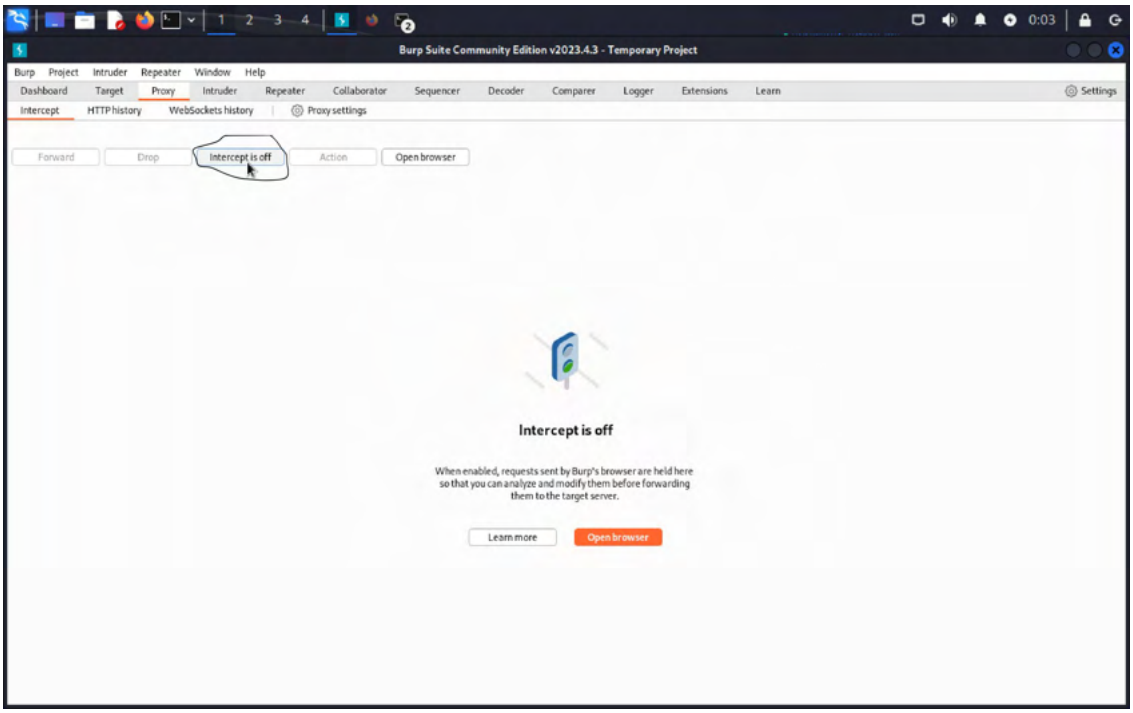


- Set your browser to use Burp as a proxy. Configure the proxy settings to point to Burp's proxy listener (default is 127.0.0.1:8080).



Enable Interception:

- In the Proxy tab, click on the "Intercept is off" button to toggle interception on.
- Intercepted requests will be displayed, and you can choose to forward, drop, or modify them.



Review Traffic:

- Use the HTTP History tab to review all intercepted requests and responses.
- Filter the history to focus on specific URLs or methods.

Modify Requests:

- In the Intercept tab, modify requests before forwarding them to the server.
- Make changes such as parameter manipulation or header modifications.

Configure Scope:

- Set the testing scope in the Scope tab to include or exclude specific URLs or domains.
- This helps focus testing on relevant areas of the application.

WebSocket and HTTP/2:

- Enable WebSocket and HTTP/2 support in the Proxy options if your application uses these protocols.
- Monitor and intercept WebSocket messages or HTTP/2 traffic.

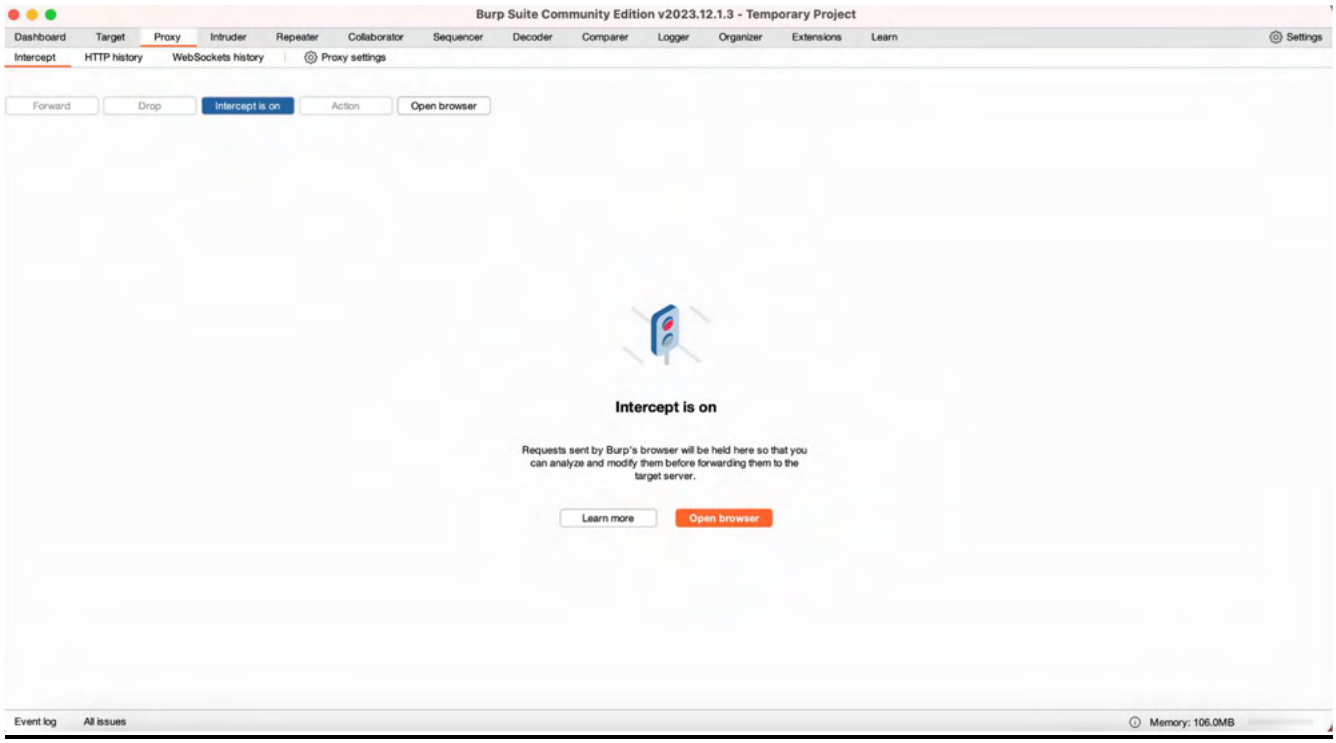
Options Configuration:

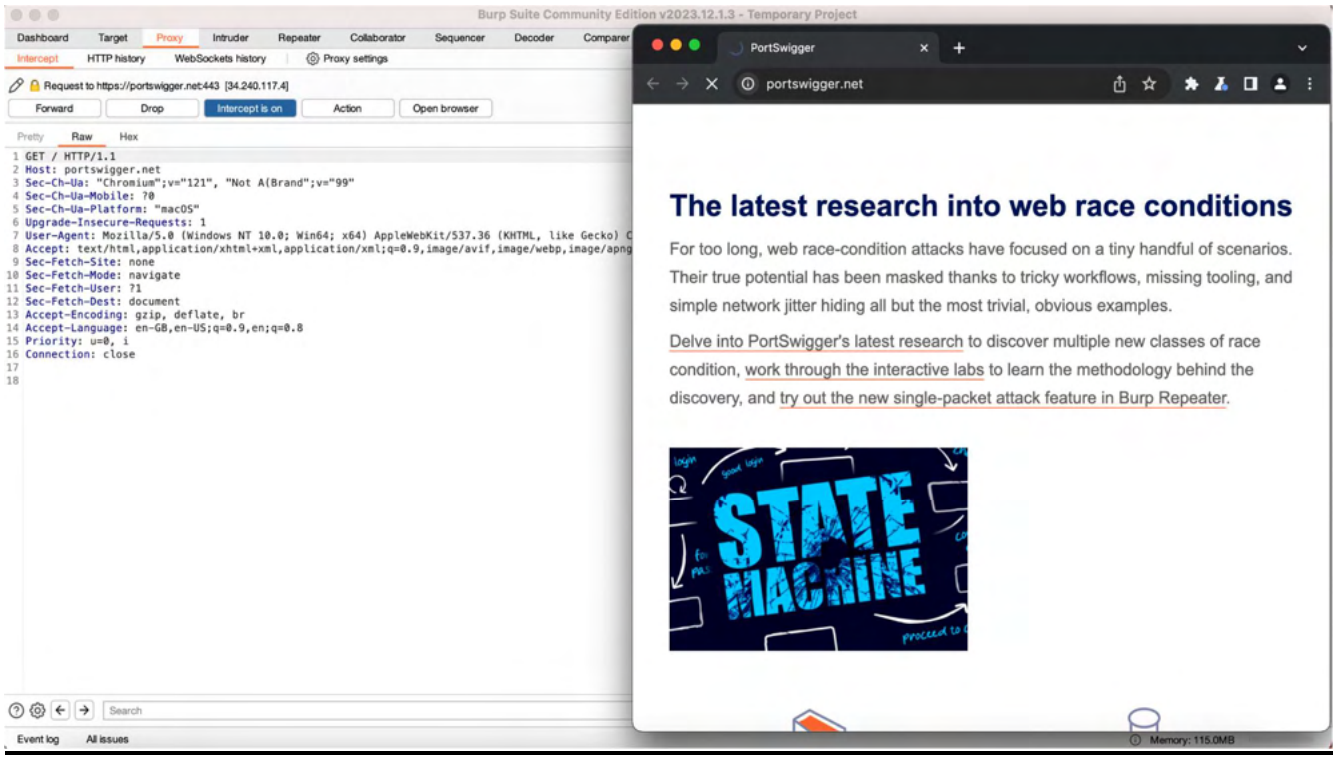
- Access the Options sub-tab to configure advanced settings for the proxy.
- Customize listeners, interception rules, and other options based on your testing needs.

Using Burp Suite's Proxy effectively is crucial for identifying security vulnerabilities in web applications. It provides a centralized point for inspecting and manipulating traffic, ensuring a thorough analysis of communication between the browser and the target application.

Intercept HTTP traffic with Burp Proxy:

-> proxy -> Intercept on -> Open browser ->





Burp Suite Community Edition v2023.12.1.3 - Temporary Project

Dashboard | Target | Proxy | Intruder | Repeater | Collaborator | Sequencer | Decoder | Comparer | Logger | Organizer | Extensions | Learn | Settings

Intercept | HTTP history | WebSockets history | Proxy settings

Request to https://portswigger.net:443 [34.240.117.4]

Forward | Drop | Intercept is on | Action | Open browser | Add notes

Pretty | Raw | Hex

```
1 GET / HTTP/1.1
2 Host: portswigger.net
3 Sec-Ch-Ua: "Chromium";v="121", "Not A(Brand";v="99"
4 Sec-Ch-Ua-Mobile: ?0
5 Sec-Ch-Ua-Platform: "macOS"
6 Upgrade-Insecure-Requests: 1
7 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/121.0.6167.85 Safari/537.36
8 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
9 Sec-Fetch-Site: none
10 Sec-Fetch-Mode: navigate
11 Sec-Fetch-User: ?1
12 Sec-Fetch-Dest: document
13 Accept-Encoding: gzip, deflate, br
14 Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
15 Priority: u=0, i
16 Connection: close
17
18
```

Inspector

Request attributes	2
Request query parameters	0
Request body parameters	0
Request cookies	0
Request headers	15

Search | 0 highlights

Event log | All issues | Memory: 115.0MB

-> forward

Burp Suite Community Edition v2023.12.1.3 - Temporary Project

Dashboard | Target | Proxy | Intruder | Repeater | Collaborator | Sequencer | Decoder | Comparer | Logger | Organizer | Extensions | Learn | Settings

Intercept | HTTP history | WebSockets history | Proxy settings

Request to https://portswigger.net:443 [34.249.63.188]

Forward | Drop | Intercept is on | Action | Open browser | Add notes | HTTP/2

Pretty | Raw | Hex

```
1 GET / HTTP/2
2 Host: portswigger.net
3 Cookie: SessionId=
  CfDJ8PtiYiF0fo5EnPHMXdJ6ToXMZuwIV9lWaw85hFYeylXLETLNgiMuMC0Pn6oMerrpWgHgr0FjeyQ%2FuJ7ggEdSVGEyqjk7VYeN2kP9v%2FuPQoPXgwxKQRiD0AFagy6kRBh4b3WbkI6b6qLsMxPws
  o1aH7A1Wb4Jhgfx1elWiTHJ%2FJh5; AWSALBAPP-0=_remove_; AWSALBAPP-1=_remove_; AWSALBAPP-2=_remove_; AWSALBAPP-3=_remove_
4 Upgrade-Insecure-Requests: 1
5 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/121.0.6167.85 Safari/537.36
6 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
7 Sec-Fetch-Site: none
8 Sec-Fetch-Mode: navigate
9 Sec-Fetch-User: ?1
10 Sec-Fetch-Dest: document
11 Sec-Ch-Ua: "Chromium";v="121", "Not A(Brand";v="99"
12 Sec-Ch-Ua-Mobile: ?0
13 Sec-Ch-Ua-Platform: "macOS"
14 Accept-Encoding: gzip, deflate, br
15 Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
16 Priority: u=0, i
17
18
```

Inspector

Request attributes	2
Request query parameters	0
Request body parameters	0
Request cookies	5
Request headers	22

Search | 0 highlights

Event log | All issues | Memory: 127.8MB

➢ Intruder Tab:

Burp Intruder is a powerful tool for performing highly customizable, automated attacks against websites.

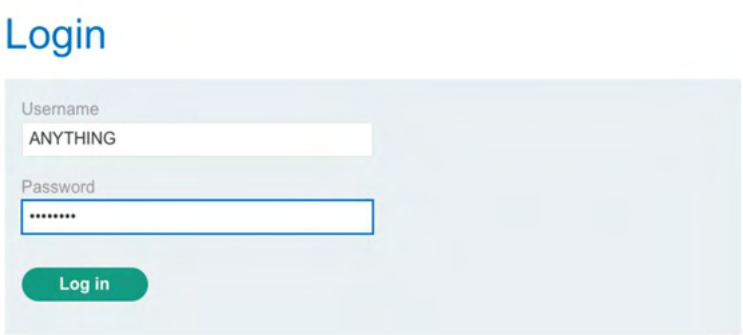
- Open Burp's browser, and use it to access the following URL:

https://portswigger.net/web-security/authentication/password-based/lab-username-enumeration-via-different-responses

Click **Access the lab** and log in to your PortSwigger account if prompted. This opens your own instance of a deliberately vulnerable blog website.

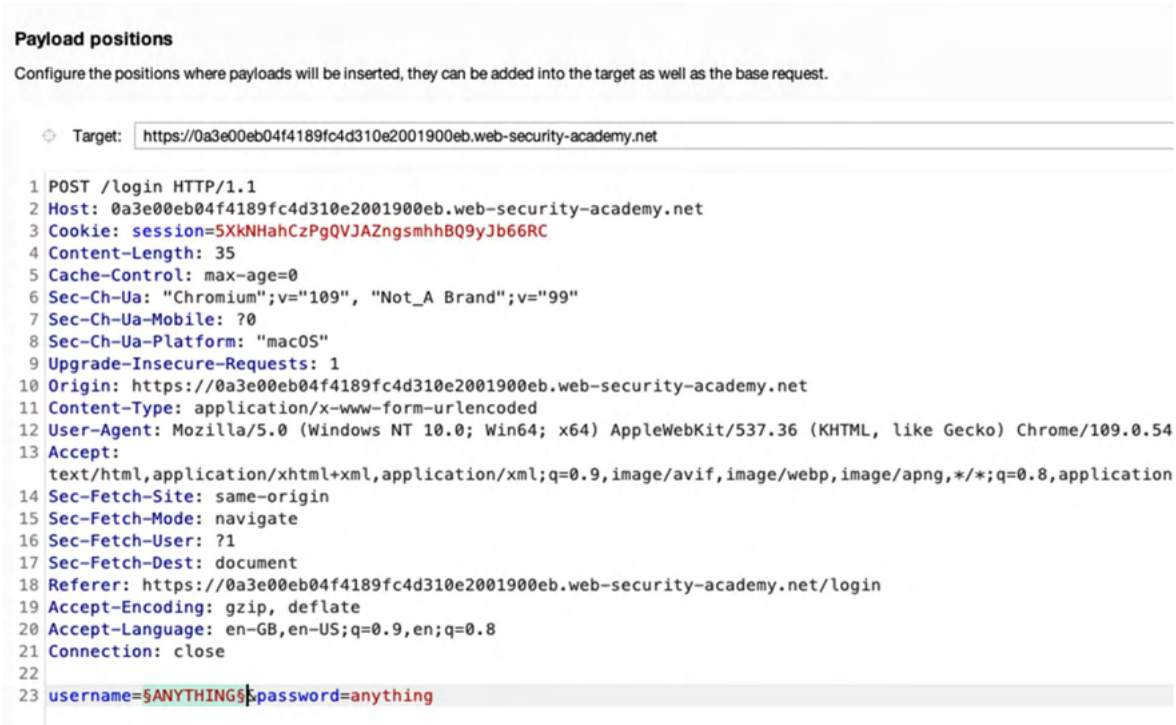
- **Try to log in**

Click **My account**, then try to log in using an invalid username and password.



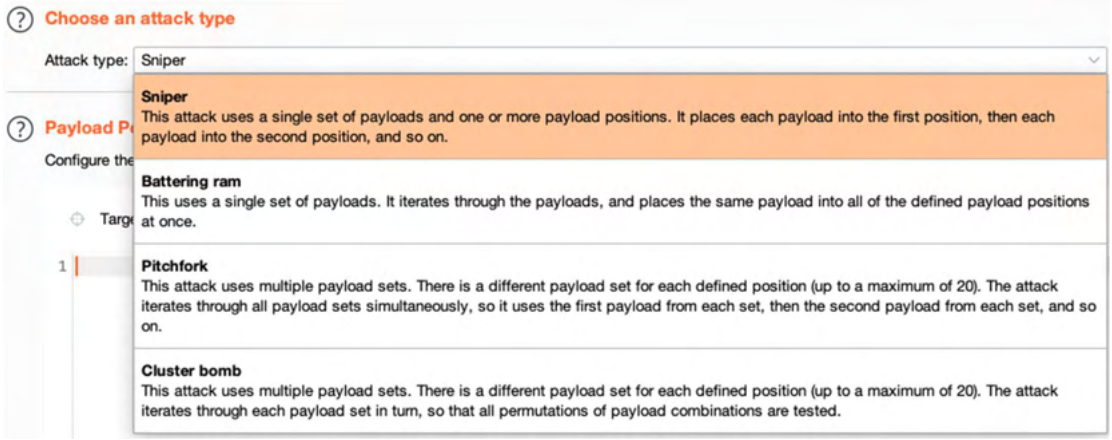
- Go to the Intruder tab. Observe that there is now a tab displaying the POST /login request. We'll use this as the base request for our attack.

Notice that the value of the username parameter that you previously highlighted is now marked as a payload position. This is indicated by the § characters at the beginning and end of the value. Burp Intruder will insert payloads at this position during the attack.



- **Select an attack type**

At the top of the screen, you can select different attack types. For now, just make sure this is set to **Sniper**.



- **Add the payloads**

You now just need to configure the list of payloads that you want to use. For this demonstration, we'll try sending the request with different usernames to test how the login mechanism behaves.

Copy the following list of candidate usernames:

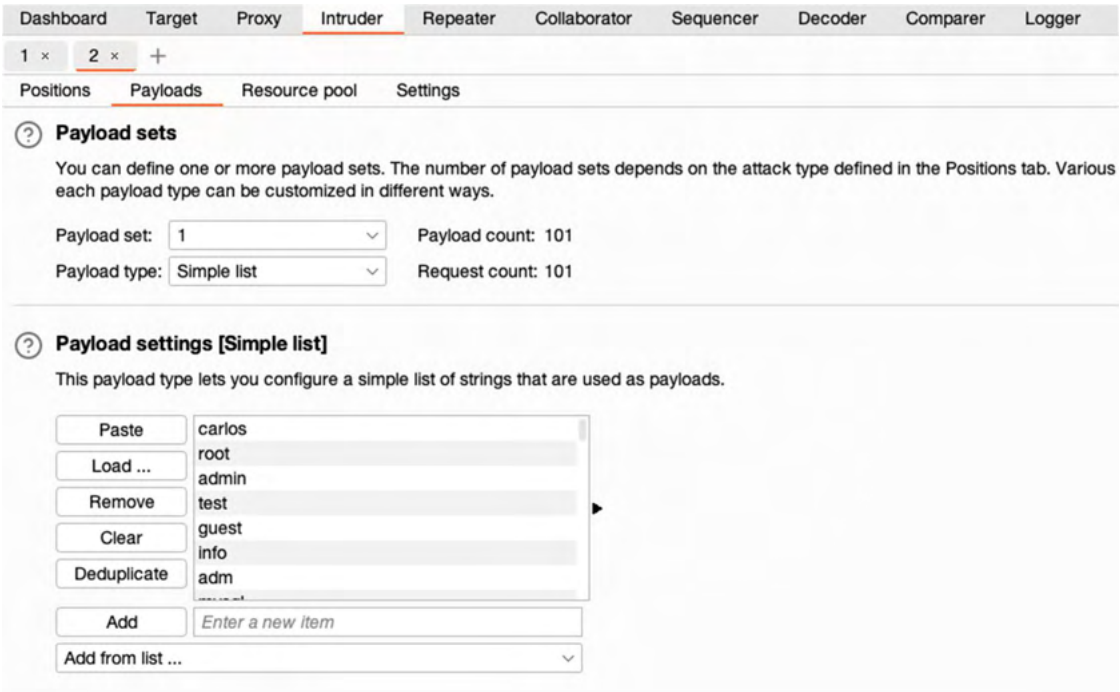
- Candidate usernames

Go to the **Payloads** tab.

Leave the **Payload type** set to **Simple list**.

In the **Payload settings** field, click **Paste** to add the copied usernames to the list.

In the **Payload sets** section, you can see how many payloads you have added, and how many requests this attack will send. For this attack, you should see: Payload count: 101 / Request count: 101.



- **Start the attack**

In the upper-right corner, click **Start attack**. This opens a new attack window in which you can see each of the requests that Burp Intruder is making.

If you select one of the entries in the table, you can view the request and response in the message editor. Notice that the username parameter contains a different value from our payload list in each request.

Results Positions Payloads Resource pool Settings

Filter: Showing all items

Request	Payload	Status	Error	Timeout	Length	Comment
4	test	200			2984	
5	guest	200			2984	
6	info	200			2984	
7	adm	200			2984	
8	mysql	200			2984	
9	user	200			2984	
10	administrator	200			2984	
11	oracle	200			2984	
12	ftp	200			2984	
13	pi	200			2984	

Request Response

Pretty Raw Hex

```
12 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/109.0.541
13 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,appl
14 Sec-Fetch-Site: same-origin
15 Sec-Fetch-Mode: navigate
16 Sec-Fetch-User: ?1
17 Sec-Fetch-Dest: document
18 Referer: https://0a3e00eb04f4189fc4d310e2001900eb.web-security-academy.net/login
19 Accept-Encoding: gzip, deflate
20 Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
21 Connection: close
22
23 username=test&password=anything
```

- **Look for any irregular responses**

The attack window contains several columns displaying key information about each response.

Wait for the attack to finish, then click the heading of the **Length** column to sort the results. As you can see, one of the responses is a different length.

Results Positions Payloads Resource pool Settings

Filter: Showing all items

Request	Payload	Status	Error	Timeout	Length	Comment
41	affiliates	200			2986	
0		200			2984	
1	carlos	200			2984	
2	root	200			2984	
3	admin	200			2984	
4	test	200			2984	
5	guest	200			2984	
6	info	200			2984	
7	adm	200			2984	
8	mysql	200			2984	
9	user	200			2984	
10	administrator	200			2984	

➢ Study the response

Select any request from the list to display it in the message editor.

Studying the responses, notice that most contain an Invalid username error message, but the one with the different length response has an Incorrect password error message.

This different response strongly suggests that this username might be valid in this case.

Results | Positions | Payloads | Resource pool | Settings

Filter: Showing all items

Request	Payload	Status	Error	Timeout	Length	Comment
41	affiliates	200			2986	
0		200			2984	
1	carlos	200			2984	
2	root	200			2984	
3	admin	200			2984	

Request | Response

Pretty | Raw | Hex | Render

```
        <p>
          |
        </p>
47    </section>
48  </header>
49  <header class="notification-header">
50  </header>
51  <h1>
      Login
    </h1>
52  <section>
53    <p class=is-warning>
        Incorrect password
      </p>
54    <form class=login-form method=POST action=/login>
55      <label>
          Username
```

Now that you have a potentially correct username, the next logical step is to try to brute-force the password. Try repeating this attack, using the username you have identified and this list of candidate passwords.

Positions | Payloads | Resource pool | Settings

Choose an attack type

Attack type: Sniper

Payload positions

Configure the positions where payloads will be inserted, they can be added into the target as well as the base request.

Target: https://0a3e00eb04f4189fc4d310e2001900eb.web-security-academy.net

```
1  POST /login HTTP/1.1
2  Host: 0a3e00eb04f4189fc4d310e2001900eb.web-security-academy.net
3  Cookie: session=5XkNHahCzPgQVJAZngsmhhBQ9yJb66RC
4  Content-Length: 35
5  Cache-Control: max-age=0
6  Sec-Ch-Ua: "Chromium";v="109", "Not_A Brand";v="99"
7  Sec-Ch-Ua-Mobile: ?0
8  Sec-Ch-Ua-Platform: "macOS"
9  Upgrade-Insecure-Requests: 1
10 Origin: https://0a3e00eb04f4189fc4d310e2001900eb.web-security-academy.net
11 Content-Type: application/x-www-form-urlencoded
12 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/109.
13 Accept:
   text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,applica
14 Sec-Fetch-Site: same-origin
15 Sec-Fetch-Mode: navigate
16 Sec-Fetch-User: ?1
17 Sec-Fetch-Dest: document
18 Referer: https://0a3e00eb04f4189fc4d310e2001900eb.web-security-academy.net/login
19 Accept-Encoding: gzip, deflate
20 Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
21 Connection: close
22
23 username=affiliates&password=§anything§
```

➢ Target Tab:

The "Target" tab in Burp Suite is a crucial component that allows users to manage and control the scope of their testing. The Target tool enables you to define which targets are in scope for your current work. It also contains the site map and **Crawl paths** tab, which show you detailed information about your target applications. Here's a detailed overview of the Target tab:

Target Tab Overview:

Scope:

Functionality:

- Defines the scope of the testing by specifying the URLs and domains to include or exclude.
- Ensures that testing efforts are focused on specific areas of the application.

How to Use:

- Add target URLs to the scope by entering them in the "Include in scope" field.
- Use the "Exclude from scope" field to exclude specific URLs or domains.

Site Map:

Functionality:

- Displays a hierarchical representation of the target application's structure.
- Provides an overview of discovered pages and their relationships.

How to Use:

- Automatically populates as you navigate through the application or perform scans.
- Right-click on items to perform actions such as adding to scope or launching scans.

Issues:

Functionality:

- Lists and categorizes security issues discovered during testing.
- Provides detailed information about each identified vulnerability.

How to Use:

- View and filter discovered issues to prioritize and address security vulnerabilities.

Scope Control:

Functionality:

- Allows quick access to control the scope settings.
- Provides options to include or exclude specific URLs or domains on the fly.

How to Use:

- Adjust scope settings dynamically by clicking on the "Scope" button and making changes.

Export:

Functionality:

- Enables the export of the site map and discovered issues.
-

- Supports various formats, including XML and CSV.

How to Use:

- Export site maps or issues data for reporting or external analysis.

Engagement Tools:

Functionality:

- Facilitates engagement with the target application.
- Includes tools like the Spider, Scanner, and Repeater.

How to Use:

- Use engagement tools to map the application, identify vulnerabilities, and test individual requests.

How to Use Burp Suite Target Tab:

Add Targets to Scope:

- Enter target URLs in the "Include in scope" field to add them for testing.
- Use the "Exclude from scope" field to exclude specific URLs or domains.

Site Map Navigation:

- Navigate through the site map to understand the structure of the application.
- Right-click on items to perform actions like adding to scope or launching scans.

Scan Configuration:

- Before launching scans, ensure that the scope is appropriately configured.
- Adjust scope settings to focus on specific areas of the application.

Issue Review:

- Monitor the "Issues" tab for a categorized list of identified vulnerabilities.
- Prioritize and address vulnerabilities based on severity and impact.

Dynamic Scope Adjustment:

- Use the "Scope" button for dynamic adjustment of the testing scope.
- Modify scope settings on the fly to adapt to testing needs.

Export Data:

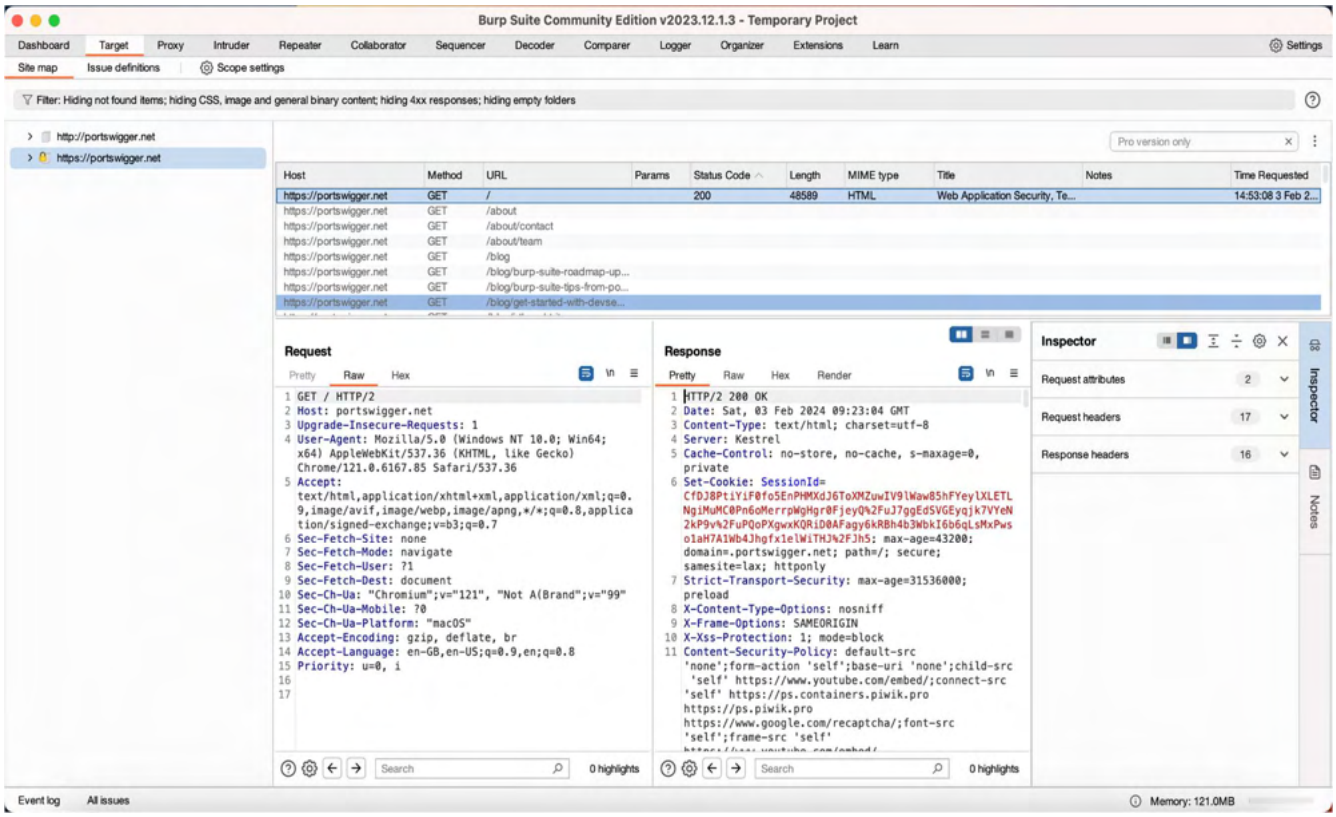
- Export the site map or issue data using the "Export" feature.
- Choose the desired format for reporting or external analysis.

Engagement Tools Integration:

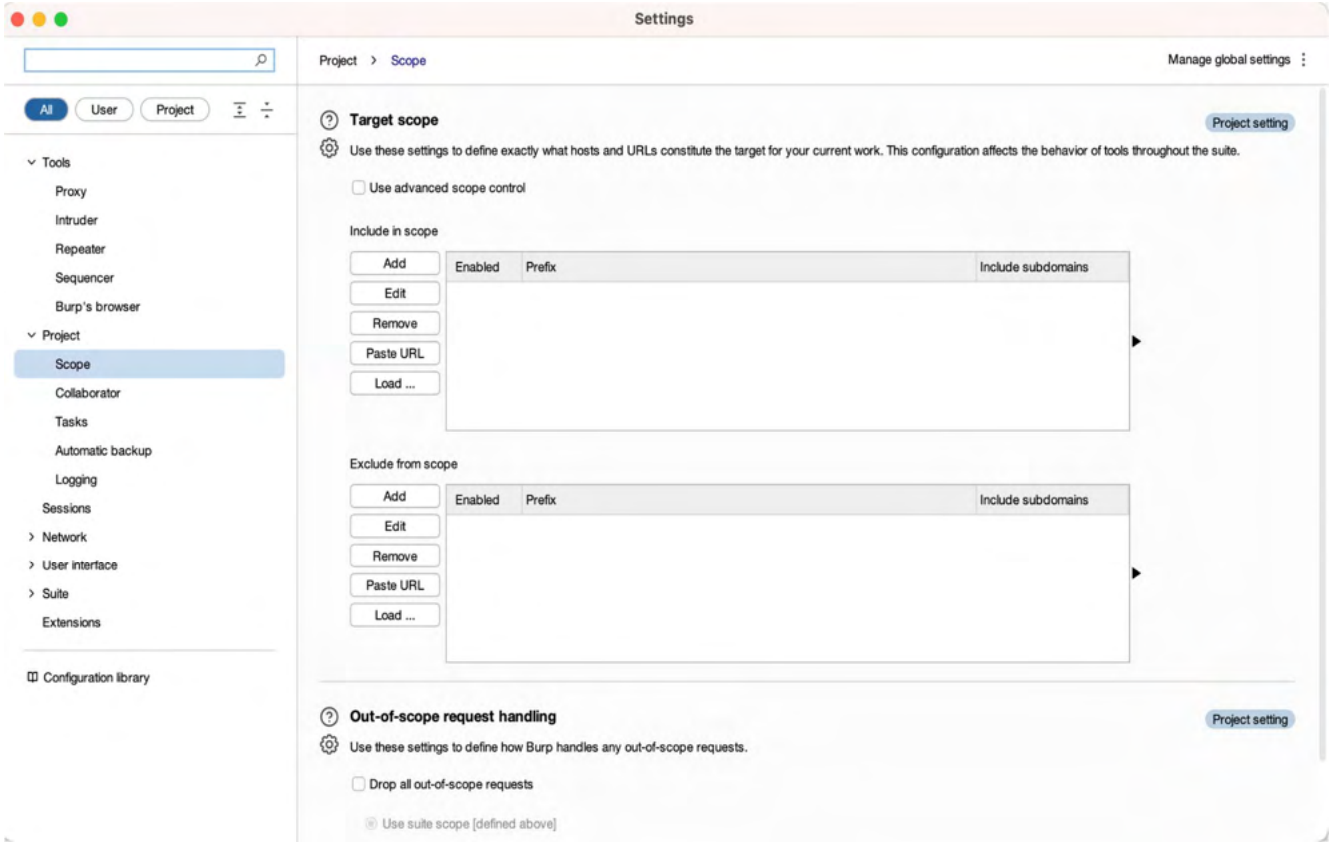
- Utilize engagement tools like Spider and Scanner directly from the Target tab.
- Perform comprehensive testing using tools available in the Burp Suite ecosystem.

The "Target" tab in Burp Suite serves as the control centre for managing the scope of your security testing efforts. By effectively using the features within this tab, you can ensure a

focused and thorough examination of the target application, identifying and addressing potential security issues.



Target scope settings:



Site map:

Items added to site map View site map

Host	Meth...	URL	Status ...	MIME Ty...
portswigger.net	GET	/	301	
portswigger.net	GET	/	200	HTML
portswigger.net		/content/images/logos/port...		
portswigger.net		/web-security		
portswigger.net		/research		
portswigger.net		/web-security/certification		
portswigger.net		/blog		
portswigger.net		/blog/burp-suite-tips-from-...		
portswigger.net		/blog/get-started-with-devs...		
portswigger.net		/blog/i-thought-it-was-a-co...		
portswigger.net		/blog/train-the-basics-bug-...		
portswigger.net		/about/team		
portswigger.net		/research/top-10-web-hac...		
portswigger.net		/web-security/getting-started		
portswigger.net		/blog/burp-suite-roadmap-...		
portswigger.net		/content/images/logos/favi...		
portswigger.net		/content/images/logos/appl...		
portswigger.net	GET	/content/pslandingpages.css	200	CSS
portswigger.net	GET	/content/fonts/ps-icons-sm...	200	woff
portswigger.net	GET	/content/fonts/ps-main/ps-i...	200	woff
portswigger.net		/users		
portswigger.net		/customers		
portswigger.net		/about		
portswigger.net		/careers		
portswigger.net		/legal		
portswigger.net		/contact		
portswigger.net		/support/reseller-faqs		
portswigger.net		/users/youraccount		
portswigger.net		/burp/enterprise		
portswigger.net	GET	/content/images/svg/icons/...	200	XML
portswigger.net		/burp/pro		
portswigger.net	GET	/content/images/svg/icons/...	200	XML

Issue Definitions:

Burp Suite Community Edition v2023.12.1.3 - Temporary Project

Dashboard | Target | Proxy | Intruder | Repeater | Collaborator | Sequencer | Decoder | Comparer | Logger | Organizer | Extensions | Learn | Settings

Site map | Issue definitions | Scope settings

Issue definitions Open in browser

This listing contains the definitions of all issues that can be detected by Burp Scanner.

Name	Typical severity	Type index
OS command injection	High	0x00100100
SQL injection	High	0x00100200
SQL injection (second order)	High	0x00100210
ASP.NET tracing enabled	High	0x00100280
File path traversal	High	0x00100300
XML external entity injection	High	0x00100400
LDAP injection	High	0x00100500
XPath injection	High	0x00100600
XML injection	Medium	0x00100700
ASP.NET debugging enabled	Medium	0x00100800
Broken Access Control	Information	0x00100850
HTTP PUT method is enabled	High	0x00100900
Out-of-band resource load (HTTP)	High	0x00100a00
File path manipulation	High	0x00100b00
PHP code injection	High	0x00100c00
Server-side JavaScript code injection	High	0x00100d00
Perl code injection	High	0x00100e00
Ruby code injection	High	0x00100f00
Python code injection	High	0x00100f10
Expression Language injection	High	0x00100f20
Unidentified code injection	High	0x00101000
Server-side template injection	High	0x00101080
SSI injection	High	0x00101100
Cross-site scripting (stored)	High	0x00200100
HTTP request smuggling	High	0x00200140
Client-side desync	High	0x00200141
Web cache poisoning	High	0x00200180
HTTP response header injection	High	0x00200200
Cross-site scripting (reflected)	High	0x00200300
Client-side template injection	High	0x00200308
Cross-site scripting (DOM-based)	High	0x00200310
Cross-site scripting (reflected DOM-based)	High	0x00200311
Cross-site scripting (stored DOM-based)	High	0x00200312
Client-side prototype pollution	Information	0x00200316
JavaScript injection (DOM-based)	High	0x00200320
JavaScript injection (reflected DOM-based)	High	0x00200321
JavaScript injection (stored DOM-based)	High	0x00200322
Path-relative style sheet import	Information	0x00200328
Client-side SQL injection (DOM-based)	High	0x00200330
Client-side SQL injection (reflected DOM-based)	High	0x00200331

OS command injection

Description

Operating system command injection vulnerabilities arise when an application incorporates user-controllable data into a command that is processed by a shell command interpreter. If the user data is not strictly validated, an attacker can use shell metacharacters to modify the command that is executed, and inject arbitrary further commands that will be executed by the server.

OS command injection vulnerabilities are usually very serious and may lead to compromise of the server hosting the application, or of the application's own data and functionality. It may also be possible to use the server as a platform for attacks against other systems. The exact potential for exploitation depends upon the security context in which the command is executed, and the privileges that this context has regarding sensitive resources on the server.

Remediation

If possible, applications should avoid incorporating user-controllable data into operating system commands. In almost every situation, there are safer alternative methods of performing server-level tasks, which cannot be manipulated to perform additional commands than the one intended.

If it is considered unavoidable to incorporate user-supplied data into operating system commands, the following two layers of defense should be used to prevent attacks:

- The user data should be strictly validated. Ideally, a whitelist of specific accepted values should be used. Otherwise, only short alphanumeric strings should be accepted. Input containing any other data, including any conceivable shell metacharacter or whitespace, should be rejected.
- The application should use command APIs that launch a specific process via its name and command-line parameters, rather than passing a command string to a shell interpreter that supports command chaining and redirection. For example, the Java API Runtime.exec and the ASP.NET API Process.Start do not support shell metacharacters. This defense can mitigate the impact of an attack even in the event that an attacker circumvents the input validation defenses.

References

- Web Security Academy: OS command injection

Vulnerability classifications

- CWE-77: Improper Neutralization of Special Elements used in a Command ('Command Injection')
- CWE-78: Improper Neutralization of Special Elements used in an OS Command ('OS Command Injection')
- CWE-116: Improper Encoding or Escaping of Output
- CAPEC-248: Command Injection

Event log | All issues | Memory: 160.3MB

Request:

Request

Pretty | Raw | Hex

```
1  GET / HTTP/2
2  Host: portswigger.net
3  Upgrade-Insecure-Requests: 1
4  User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64;
   x64) AppleWebKit/537.36 (KHTML, like Gecko)
   Chrome/121.0.6167.85 Safari/537.36
5  Accept:
   text/html,application/xhtml+xml,application/xml;q=0.
   9,image/avif,image/webp,image/apng,*/*;q=0.8,applica
   tion/signed-exchange;v=b3;q=0.7
6  Sec-Fetch-Site: none
7  Sec-Fetch-Mode: navigate
8  Sec-Fetch-User: ?1
9  Sec-Fetch-Dest: document
10 Sec-Ch-Ua: "Chromium";v="121", "Not A(Brand";v="99"
11 Sec-Ch-Ua-Mobile: ?0
12 Sec-Ch-Ua-Platform: "macOS"
13 Accept-Encoding: gzip, deflate, br
14 Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
15 Priority: u=0, i
16
17
```

Search | 0 highlights

Response:

Response

Pretty | Raw | Hex | Render

```
1  HTTP/2 200 OK
2  Date: Sat, 03 Feb 2024 09:23:04 GMT
3  Content-Type: text/html; charset=utf-8
4  Server: Kestrel
5  Cache-Control: no-store, no-cache, s-maxage=0,
   private
6  Set-Cookie: SessionId=
   CfDJ8PtiYiF0fo5EnPHMXdJ6ToXMZuwIV9lWaw85hFYeylXLETL
   NgiMuMC0Pn6oMerrpWgHgr0FjeyQ%2FuJ7ggEdSVGEyqjk7VYeN
   2kP9v%2FuPQoPXgwxKQRiD0AFagy6kRBh4b3WbkI6b6qLsMxPws
   o1aH7A1Wb4Jhgfx1elWiTHJ%2FJh5; max-age=43200;
   domain=.portswigger.net; path=/; secure;
   samesite=lax; httponly
7  Strict-Transport-Security: max-age=31536000;
   preload
8  X-Content-Type-Options: nosniff
9  X-Frame-Options: SAMEORIGIN
10 X-Xss-Protection: 1; mode=block
11 Content-Security-Policy: default-src
   'none';form-action 'self';base-uri 'none';child-src
    'self' https://www.youtube.com/embed/;connect-src
   'self' https://ps.containers.piwik.pro
   https://ps.piwik.pro
   https://www.google.com/recaptcha/;font-src
   'self';frame-src 'self'
```

Search | 0 highlights

Response

Pretty | Raw | Hex | Render

```
00000000  48 54 54 50 2f 32 20 32   30 30 20 4f 4b 0d 0
00000010  61 74 65 3a 20 53 61 74   2c 20 30 33 20 46 6
00000020  20 32 30 32 34 20 30 39   3a 32 33 3a 30 34 2
00000030  4d 54 0d 0a 43 6f 6e 74   65 6e 74 2d 54 79 7
00000040  3a 20 74 65 78 74 2f 68   74 6d 6c 3b 20 63 6
00000050  72 73 65 74 3d 75 74 66   2d 38 0d 0a 53 65 7
00000060  65 72 3a 20 4b 65 73 74   72 65 6c 0d 0a 43 6
00000070  68 65 2d 43 6f 6e 74 72   6f 6c 3a 20 6e 6f 2
00000080  74 6f 72 65 2c 20 6e 6f   2d 63 61 63 68 65 2
00000090  73 2d 6d 61 78 61 67 65   3d 30 2c 20 70 72 6
000000a0  61 74 65 0d 0a 53 65 74   2d 43 6f 6f 6b 69 6
000000b0  20 53 65 73 73 69 6f 6e   49 64 3d 43 66 44 4
000000c0  50 74 69 59 69 46 30 66   6f 35 45 6e 50 48 4
000000d0  64 4a 36 54 6f 58 4d 5a   75 77 49 56 39 6c 5
000000e0  77 38 35 68 46 59 65 79   6c 58 4c 45 54 4c 4
000000f0  69 4d 75 4d 43 30 50 6e   36 6f 4d 65 72 72 7
00000100  67 48 67 72 30 46 6a 65   79 51 25 32 46 75 4
00000110  67 67 45 64 53 56 47 45   79 71 6a 6b 37 56 5
00000120  4e 32 6b 50 39 76 25 32   46 75 50 51 6f 50 5
00000130  77 78 4b 51 52 69 44 30   41 46 61 67 79 36 6
00000140  42 68 34 62 33 57 62 6b   49 36 62 36 71 4c 7
00000150  78 50 77 73 6f 31 61 48   37 41 31 57 62 34 4
00000160  67 66 78 31 65 6c 57 69   54 48 4a 25 32 46 4
00000170  35 3b 20 6d 61 78 2d 61   67 65 3d 34 33 32 3
00000180  3b 20 64 6f 6d 61 69 6e   3d 2e 70 6f 72 74 7
```