NAME: NANDIGAM KAMALI HARIPRIYA
REG NO: 21BCE2746
AI for Cyber Security with IBM Qradar (AI for Web Security)

Assignment - 4: Working with Burp Suite

What is a Burp Suite?

Burp Suite is a set of tools developed by Portswigger, also known as Dafydd Stuttard, for web application penetration testing. It is an all-in-one tool that can be enhanced with BApps add-ons. Burp is popular among professional web app security researchers and bug bounty hunters due to its ease of use. It is available in a community edition for free, a professional edition for $399/year, and an enterprise edition for $3999/year. This article provides a brief introduction to BurpSuite, and beginners should read through without too much detail.

Why Burp Suite?

Burp Suite is a widely used cybersecurity tool designed for web application security testing and analysis. It helps security professionals, penetration testers, and developers identify vulnerabilities and security issues in web applications, such as SQL injection, cross-site scripting (XSS), and Cross-Site Request Forgery. Burp Suite offers both manual and automated scanning capabilities, allowing security professionals to manually explore web applications, analyze requests and responses, and manipulate data to find vulnerabilities.

What are the features of burp suite?

1. Spider: Spider is a web spider/crawler used to map a web application, gathering endpoints for functionality observation and potential vulnerabilities. It is used to gather more endpoints during recon processes, as the more they gather, the more attack surfaces are present during testing.
2. Proxy: BurpSuite features an intercepting proxy that allows users to view and modify request and response content during transit. It also allows users to send monitored requests to other tools, eliminating the need for copy-paste. The proxy server can be configured to run on specific loop-back IPs and ports, and filter out specific types of request-response pairs.
3. Intruder: The intruder is a fuzzer used to run a set of values through an input point, observing the output for success/failure and content length. BurpSuite allows brute-force, dictionary file, and single values for its payload position. It is used for brute-force attacks on password forms, pin forms, and fields suspected of vulnerability to XSS or SQL injection, as well as testing and attacking rate limiting on the web-app.
4. Repeater: A repeater is a tool that allows users to send requests repeatedly with manual modifications. It is used to verify user-supplied values, understand the server's expectations in input parameters, handle unexpected values, and ensure input sanitation. It also helps identify the session cookie and CSRF protection. The server's sanitation style and the actual session cookie are also important aspects to consider. It's essential to understand how to use a repeater effectively.
5. Sequencer: The sequencer is an entropy checker that checks the randomness of tokens generated by web servers, such as cookies and anti-CSRF tokens, for authentication in sensitive operations. To achieve uniform distribution of character appearances, tokens should be generated in a fully random manner. An entropy analyzer tests this hypothesis by assuming tokens are random and testing them on certain parameters for

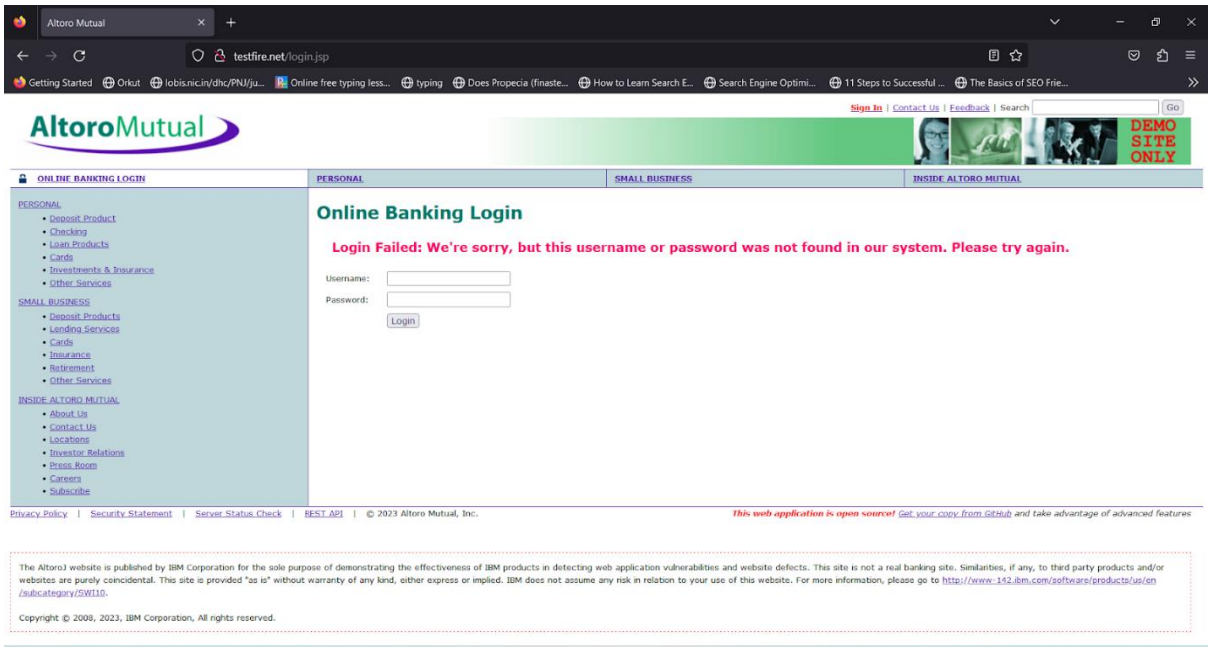
certain characteristics. A significance level is defined as a minimum probability value for a characteristic, and if the token's probability is below this level, the hypothesis that the token is random is rejected. This tool can identify weak tokens and enumerate their construction.

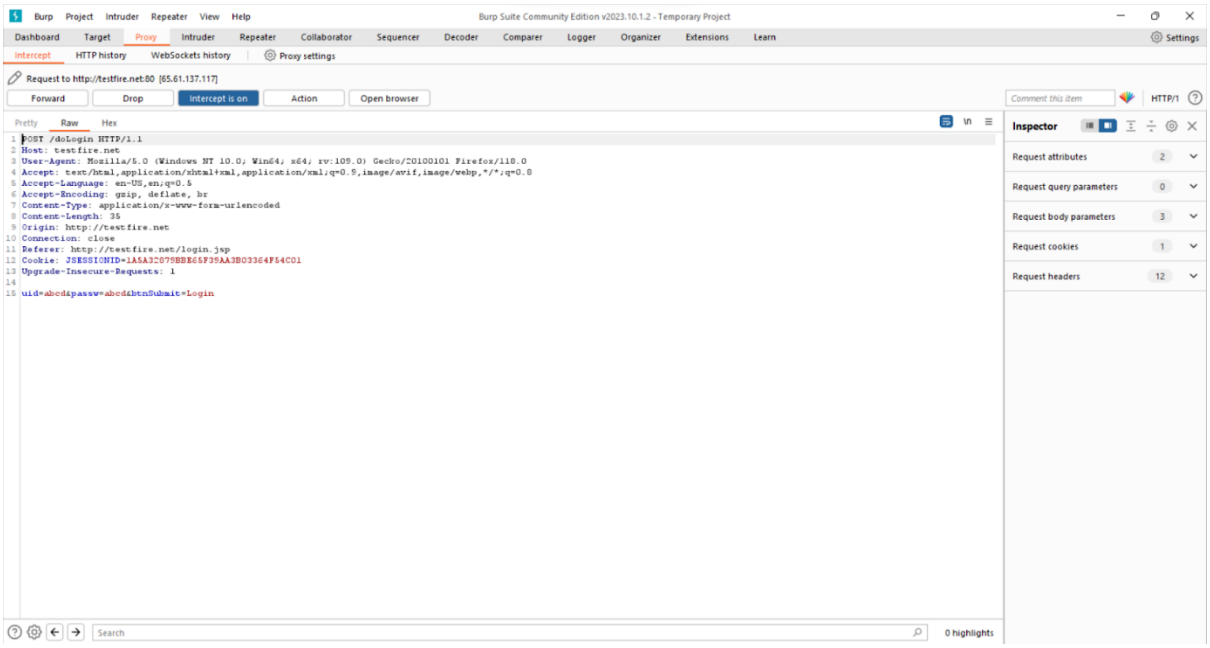
6. Decoder: The BurpSUite Decoder is a tool that decodes data using common encoding methods like URL, HTML, Base64, and Hex, aiding in data analysis, payload construction, and identifying primary cases of IDOR and session hijacking, particularly in parameters or header values.
7. Extender: BurpSuite integrates external components called BApps into its tools suite, enhancing its capabilities. These BApps function like browser extensions and can be viewed, modified, installed, or uninstalled in the Extender window. Some are supported on the community version, while others require the paid professional version.
8. Scanner: The scanner, not available in the community edition, automatically scans websites for common vulnerabilities, lists them with confidence and exploitation complexity, and is regularly updated to include new and less known vulnerabilities.

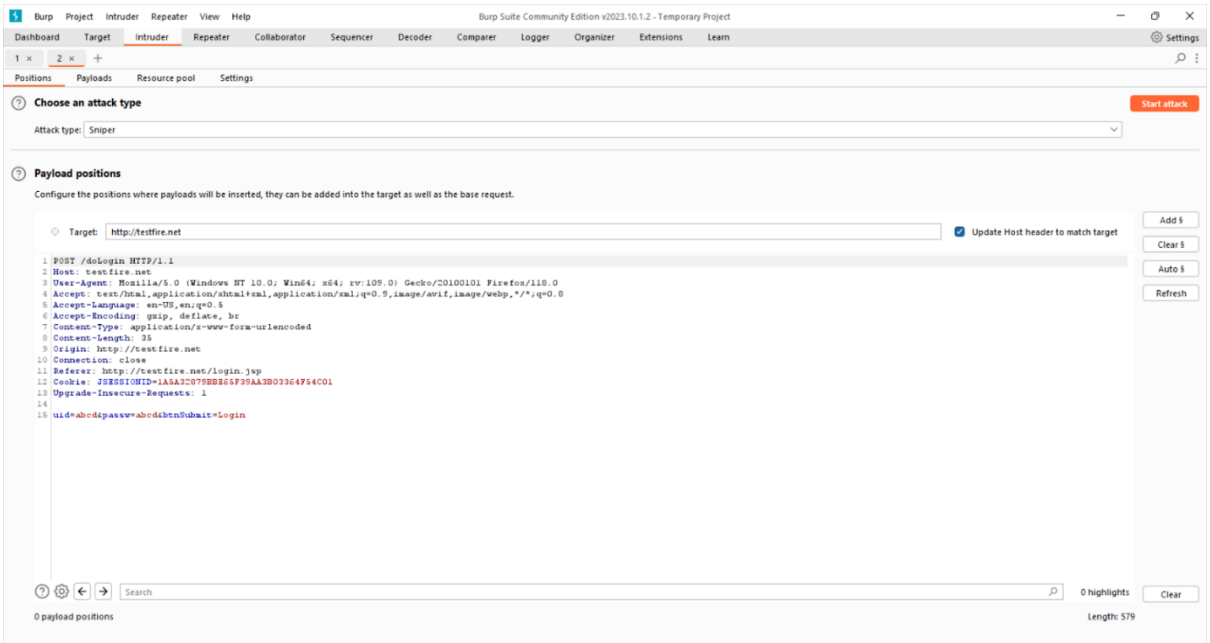
Source:
https://www.geeksforgeeks.org/what-is-burp-suite/

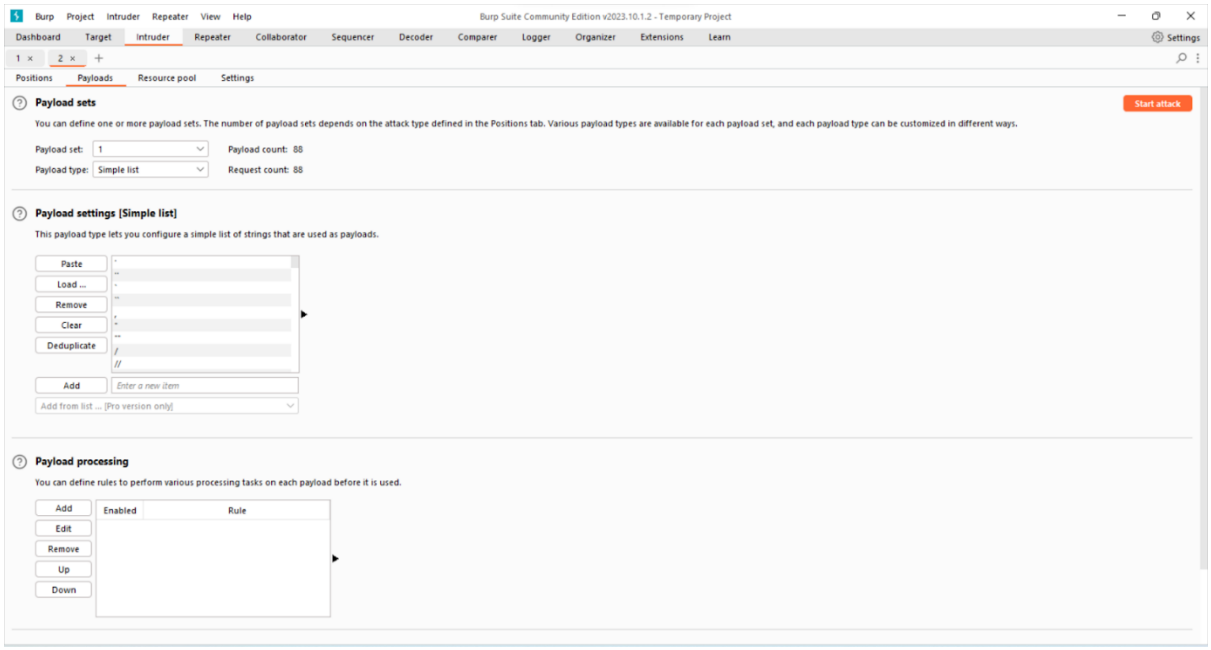
Test the vulnerabilities of testfire.net

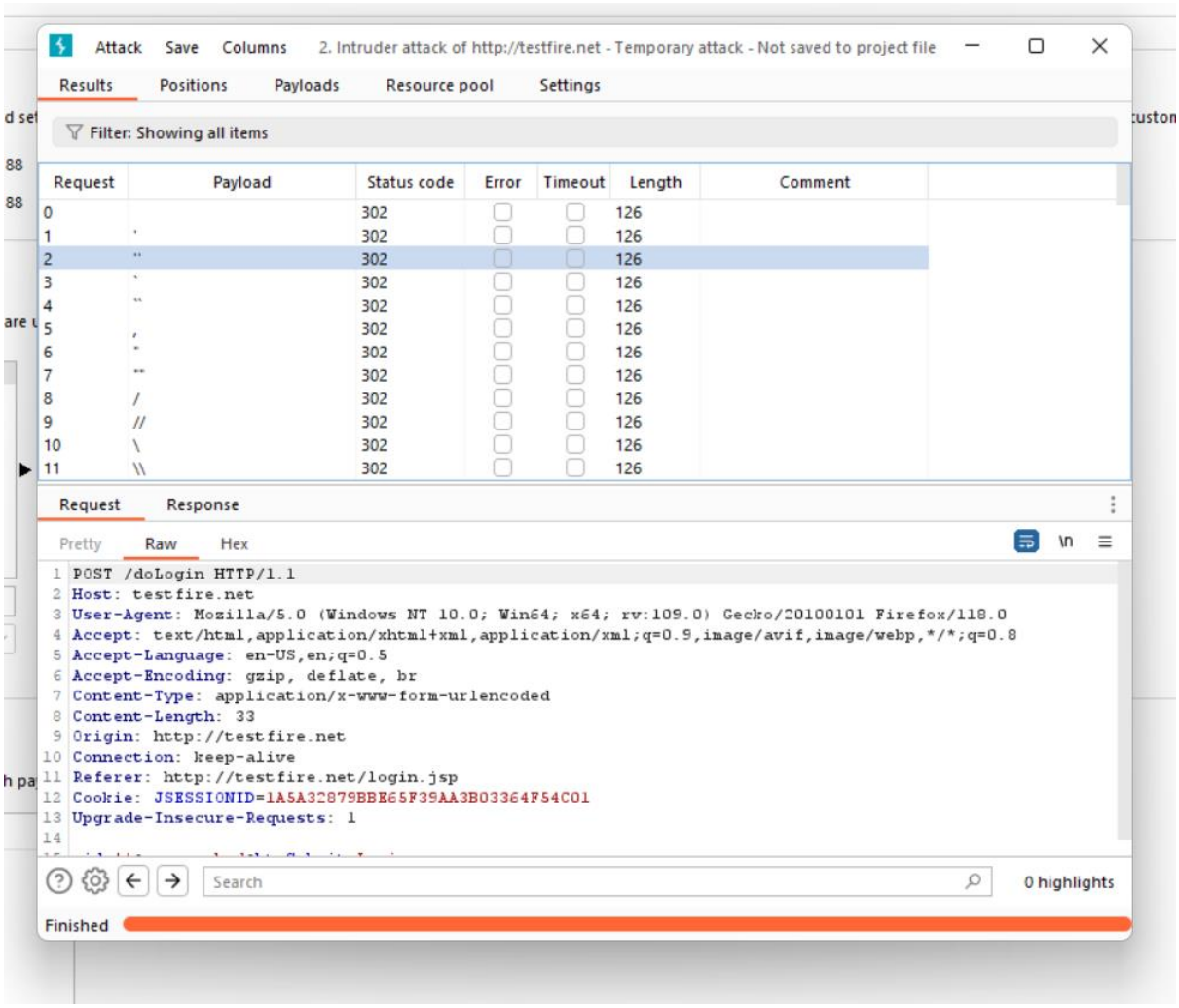
http://testfire.net

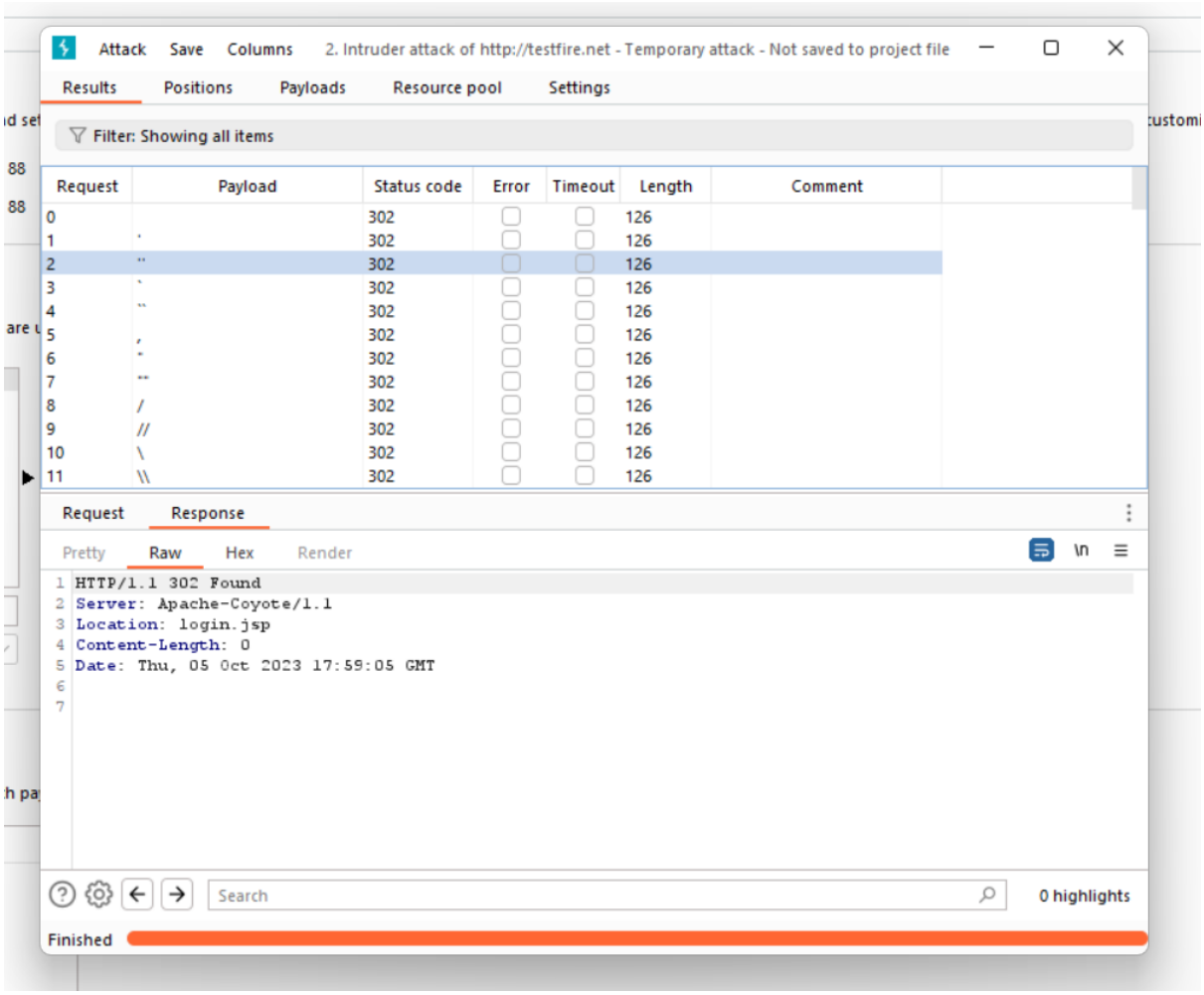


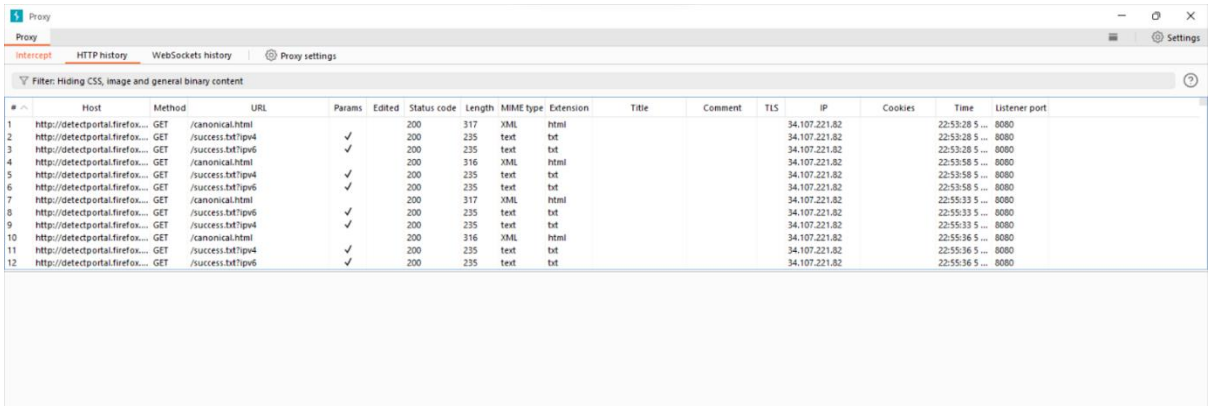

Burp Project Intruder Repeater View Help
Burp Suite Community Edition v2023.10.1.2 - Temporary Project
Dashboard Target Proxy Intruder Repeater Collaborator Sequencer Decoder Comparer Logger Organizer Extensions Learn
Settings
Intercept HTTP history WebSockets history Proxy settings
Request to http://testfire.net:80 [65.61.137.117]
Forward Drop Intercept is on Action Open browser
Comment this item HTTP/1
Pretty Raw Hex
1 POST /doLogin HTTP/1.1
2 Host: testfire.net
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/118.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate, br
7 Content-Type: application/x-www-form-urlencoded
8 Content-Length: 35
9 Origin: http://testfire.net
10 Connection: close
11 Referer: http://testfire.net/login.jsp
12 Cookie: JSESSIONID=1A5A32C75BBE65F39AA3B03364F54C01
13 Upgrade-Insecure-Requests: 1
14
15 uid=abcd&passw=abcd&btnSubmit=Login
Inspector
Request attributes 2
Request query parameters 0
Request body parameters 3
Request cookies 1
Request headers 12
Search
0 highlights


Burp Project Intruder Repeater View Help
Burp Suite Community Edition v2023.10.1.2 - Temporary Project
Dashboard Target Intruder Repeater Collaborator Sequencer Decoder Comparer Logger Organizer Extensions Learn
Settings
1 × 2 ×
Positions Payloads Resource pool Settings
Choose an attack type
Start attack
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: http://testfire.net
Update Host header to match target
Add §
Clear §
Auto §
Refresh
1 POST /doLogin HTTP/1.1
2 Host: testfire.net
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/118.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate, br
7 Content-Type: application/x-www-form-urlencoded
8 Content-Length: 35
9 Origin: http://testfire.net
10 Connection: close
11 Referer: http://testfire.net/login.jsp
12 Cookie: JSESSIONID=1A5A32C75BBE65F39AA3B03364F54C01
13 Upgrade-Insecure-Requests: 1
14
15 uid=abcd&passw=abcd&btnSubmit=Login
Search
0 highlights
Clear
0 payload positions
Length: 579


Burp Project Intruder Repeater View Help
Burp Suite Community Edition v2023.10.1.2 - Temporary Project
Dashboard Target Intruder Repeater Collaborator Sequencer Decoder Comparer Logger Organizer Extensions Learn
Settings
Positions Payloads Resource pool Settings
Payload sets
Start attack
You can define one or more payload sets. The number of payload sets depends on the attack type defined in the Positions tab. Various payload types are available for each payload set, and each payload type can be customized in different ways.
Payload set: 1
Payload count: 88
Payload type: Simple list
Request count: 88
Payload settings [Simple list]
This payload type lets you configure a simple list of strings that are used as payloads.
Paste
Load ...
Remove
Clear
Deduplicate
Add
Enter a new item
Add from list ... [Pro version only]
Payload processing
You can define rules to perform various processing tasks on each payload before it is used.
Add
Edit
Remove
Up
Down
Enabled
Rule


Attack Save Columns
2. Intruder attack of http://testfire.net - Temporary attack - Not saved to project file
Results Positions Payloads Resource pool Settings
Filter: Showing all items
Request Payload Status code Error Timeout Length Comment
0 302 126
1 ' 302 126
2 '' 302 126
3 ` 302 126
4 `` 302 126
5 , 302 126
6 " 302 126
7 "" 302 126
8 / 302 126
9 // 302 126
10 \ 302 126
11 \\ 302 126
Request Response
Pretty Raw Hex
1 POST /doLogin HTTP/1.1
2 Host: testfire.net
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/118.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate, br
7 Content-Type: application/x-www-form-urlencoded
8 Content-Length: 33
9 Origin: http://testfire.net
10 Connection: keep-alive
11 Referer: http://testfire.net/login.jsp
12 Cookie: JSESSIONID=1A5A32879BBE65F39AA3B03364F54C01
13 Upgrade-Insecure-Requests: 1
14
Search
0 highlights
Finished


Attack Save Columns 2. Intruder attack of http://testfire.net - Temporary attack - Not saved to project file
Results Positions Payloads Resource pool Settings
Filter: Showing all items
Request Payload Status code Error Timeout Length Comment
0 302 126
1 ' 302 126
2 '' 302 126
3 ` 302 126
4 `` 302 126
5 , 302 126
6 " 302 126
7 "" 302 126
8 / 302 126
9 // 302 126
10 \ 302 126
11 \\ 302 126
Request Response
Pretty Raw Hex Render
1 HTTP/1.1 302 Found
2 Server: Apache-Coyote/1.1
3 Location: login.jsp
4 Content-Length: 0
5 Date: Thu, 05 Oct 2023 17:59:05 GMT
Search 0 highlights
Finished


Proxy
Intercept HTTP history WebSockets history Proxy settings
Filter: Hiding CSS, image and general binary content
Host Method URL Params Edited Status code Length MIME type Extension Title Comment TLS IP Cookies Time Listener port
1 http://detectportal.firefox.... GET /canonical.html 200 317 XML html 34.107.221.82 22:53:28 5 ... 8080
2 http://detectportal.firefox.... GET /success.txt?ipv4 ✓ 200 235 text txt 34.107.221.82 22:53:28 5 ... 8080
3 http://detectportal.firefox.... GET /success.txt?ipv6 ✓ 200 235 text txt 34.107.221.82 22:53:28 5 ... 8080
4 http://detectportal.firefox.... GET /canonical.html 200 316 XML html 34.107.221.82 22:53:58 5 ... 8080
5 http://detectportal.firefox.... GET /success.txt?ipv4 ✓ 200 235 text txt 34.107.221.82 22:53:58 5 ... 8080
6 http://detectportal.firefox.... GET /success.txt?ipv6 ✓ 200 235 text txt 34.107.221.82 22:53:58 5 ... 8080
7 http://detectportal.firefox.... GET /canonical.html 200 317 XML html 34.107.221.82 22:55:33 5 ... 8080
8 http://detectportal.firefox.... GET /success.txt?ipv6 ✓ 200 235 text txt 34.107.221.82 22:55:33 5 ... 8080
9 http://detectportal.firefox.... GET /success.txt?ipv4 ✓ 200 235 text txt 34.107.221.82 22:55:33 5 ... 8080
10 http://detectportal.firefox.... GET /canonical.html 200 316 XML html 34.107.221.82 22:55:36 5 ... 8080
11 http://detectportal.firefox.... GET /success.txt?ipv4 ✓ 200 235 text txt 34.107.221.82 22:55:36 5 ... 8080
12 http://detectportal.firefox.... GET /success.txt?ipv6 ✓ 200 235 text txt 34.107.221.82 22:55:36 5 ... 8080

v