Team Members:

16010121195: Shreshtha Sodhia

16010121215: Vidisha Wanjare

16010121220: Zarwaan Shroff

Topic: Internal Assessment I- Implementation of any security tool

Date: 15th February, 2024

Chosen Tool: Burp Suite

INTRODUCTION

Today, we've delved into the intricate world of Burp Suite, a renowned security tool revered by professional web application security researchers and bug bounty hunters alike. As a comprehensive collection of tools tailored for testing web application security, Burp Suite stands out for its versatility and expandability.

At its core, Burp Suite serves as a Web Penetration Testing framework, setting the industry standard for information security professionals. By harnessing various Web3 technologies, it facilitates a thorough testing process, from initial mapping and analysis of an application's attack surface to the identification and exploitation of security flaws. Notably, Burp Suite excels in manual software testing, capturing and analyzing everything the browser views, facilitating efficient HTTP breaking, click-checking assault detection, and revealing concealed attack surfaces.

Functioning as an interception proxy, Burp Suite seamlessly integrates into the testing workflow by routing traffic through its proxy server, thereby enabling penetration testers to scrutinize each request to and from the target web application. This "Man In The Middle" approach allows testers to pause, manipulate, and replay individual HTTP requests, identifying potential parameters or injection points for further analysis.

Moreover, Burp Suite empowers testers with the capability to specify injection points for both manual and automated fuzzing attacks, uncovering unintended application behaviors, crashes, and error messages.

In essence, Burp Suite serves as an indispensable ally in the quest for web application security, offering a comprehensive suite of tools and features designed to verify attack vectors effectively. Its widespread adoption and robust functionality have solidified its position as an essential component in the arsenal of information security professionals worldwide.

FEATURES

In this demonstration, we delved into the Community Edition of Burp, demonstrating its vital features such as the proxy, along with a comprehensive history log. Among its essential tools that we have explored are the Repeater, Intruder, Sequencer, and Comparer, each serving crucial roles in the assessment process.

- Proxy

The Burp Proxy functions as an intermediary web server positioned between the browser and targeted applications. It empowers users to intercept, scrutinize, and manipulate the bidirectional flow of traffic.

- Repeater

Burp Repeater serves as a mechanism for iteratively modifying and dispatching intriguing HTTP or WebSocket messages.

- Intruder

Burp Intruder serves as a sophisticated tool for automating tailored assaults on web applications. It facilitates the configuration of repetitive attacks, ensuring the retransmission of identical HTTP requests.

- Sequencer

Burp Sequencer facilitates the assessment of randomness quality within a set of tokens. Users can leverage Sequencer to evaluate the unpredictability of any tokens.

- Comparer

Burp Comparer enables the comprehensive comparison of two data items. Users can utilize Comparer to swiftly identify nuanced disparities between requests or responses.

METHODOLOGY

I. Utilizing Burp Proxy:

1. Launch Burp Suite and Set Up Proxy:
   - Open Burp Suite and navigate to the dashboard.
   - Access the "Proxy" tab and toggle the intercept feature on.
2. Configure Browser to Work with Burp:
   - Click on "Open Browser" within Burp Suite.
   - The launched browser is automatically configured to work with Burp Suite.
3. Navigate to the Target Website:
   - Visit the website or web application under test in the browser.
4. Intercept HTTP Requests:
   - Observe that the site may not load initially due to the interception by Burp Proxy.
   - Access the intercepted HTTP request in the intercept tab of Burp Suite.
   - Modify the intercepted request as needed before forwarding it to the target server.
5. Forward Modified Requests:
   - After making necessary modifications, forward the intercepted request to the target server using the designated button in Burp Suite.
6. Perform Manipulations and Exploit Vulnerabilities:
   - Within the browser, perform actions such as logging in and entering credentials.
   - Utilize Burp Proxy to modify parameters or payloads in the intercepted requests.
   - Observe changes in the behavior of the web application due to these manipulations.
   - Exploit vulnerabilities, if identified, by altering request parameters or payloads.
   - Evaluate the impact of these manipulations on the functionality and security of the web application.
7. Disable Interception:
   - Once the necessary testing and manipulation are completed, switch off the interception feature in Burp Proxy.

8. Analyze Results and Report Findings:
- Examine the results of the testing, including any vulnerabilities discovered or exploited.
- Document the findings, including details of vulnerabilities, impact assessments, and recommended remediation actions.
- Prepare a comprehensive report summarizing the testing process, findings, and recommendations for stakeholders.

9. Repeat and Iterate:
- Iterate through the testing process, adjusting techniques and strategies as necessary to thoroughly assess the web application's security posture.
- Continuously monitor for new vulnerabilities and perform regular security assessments to maintain the security of the web application over time.

II. Utilizing Burp Sequencer

1. Accessing Sequencer:
- Navigate to the Proxy tab in Burp Suite.
- Select the desired HTTP request from the history, typically involving login or token generation.
- Send the selected request to the Sequencer for analysis.

2. Token Location Specification:
- In the Sequencer tab, the captured request is automatically selected.
- Specify the location of the token within the response, using either a standard form field or a custom location extraction rule, if necessary.

3. Initiate Live Capture:
- Click on "Start live capture" to instruct Burp Sequencer to repeatedly issue the request and extract tokens from the responses.

4. Analysis and Results:
- Analyze the captured tokens and their randomness properties in the Live capture window.
- Utilize the Character-level analysis and Bit-level analysis tabs to gain insights into the randomness quality and identify any anomalies.

- Review the Summary tab for a consolidated overview of the analysis results, including overall randomness estimates, effective entropy charts, and significance levels.
- Evaluate the reliability of the results based on the sample size provided by Sequencer.

5. Interpretation and Reporting:
- Interpret the results to assess the effectiveness of token generation mechanisms and the potential for token prediction attacks.
- Document the findings, including details of randomness assessments and any recommendations for improving token generation practices.

III. Utilizing Burp Comparer:

1. Selecting Items for Comparison:
- Right-click on the desired pages or responses to be compared.
- Send the selected items to the Comparer for analysis.

2. Choosing Comparison Mode:
- Navigate to the Comparer tab.
- Choose whether to compare the content of the items based on Words or Bytes.

3. Viewing Comparison Results:
- The comparison results open in a new window, displaying the compared items side by side.
- Modifications, deletions, and additions are color-coded to highlight differences between the items.
- Text not highlighted indicates content present in both items.

4. Analysis and Interpretation:
- Analyze the comparison results to identify subtle differences between the items, such as variations in responses to different inputs or requests.
- Use the identified differences to gain insights into potential security vulnerabilities or misconfigurations in the application.

IV. Utilizing Burp Repeater:

1. Capturing Requests:
   - Navigate to the target website or web application.
   - Perform actions such as logging in or submitting forms.
   - Send the requests to Burp Suite's proxy for interception.
2. Sending Requests to Repeater:
   - After capturing the desired request in the Proxy tab, right-click and select "Send to Repeater."
   - Alternatively, manually copy and paste the request into the Repeater tab.
3. Modifying and Resending Requests:
   - In the Repeater tab, modify parameters or payloads within the request as needed.
   - Click on the "Send" button to resend the modified request to the target server.
   - Analyze the response received in the Repeater tab for any changes or vulnerabilities.
4. Inspecting Response:
   - View the entire HTML structure of the response in the "Pretty" tab.
   - Preview the web page in the "Render" tab to visualize any changes made.
5. Analyzing Results:
   - Assess the impact of modifications on the application's behavior and security.
   - Identify vulnerabilities such as input validation failures or authentication bypasses.
   - Document findings and recommendations for remediation.

V. Utilizing Burp Intruder:

1. Selecting Requests for Attack:
   - Choose the target request, typically involving login or input submission, from the HTTP history.
   - Send the selected request to the Intruder tab for further analysis.
2. Configuring Attack Type:
   - Modify the values of parameters or payloads within the request as necessary, such as usernames or passwords.
   - Set the attack type based on the desired method of payload insertion: Sniper or Cluster Bomb.

3. Defining Payloads:
   - Specify the payloads to be used in the attack, such as a list of usernames or passwords.
   - Configure the payload positions within the request where the payloads will be inserted.
4. Starting the Attack:
   - Click on the "Start Attack" button to initiate the attack process.
   - Monitor the progress of the attack in the Intruder tab, observing how each payload is tested against the target.
5. Analyzing Attack Results:
   - Review the results of the attack to identify successful payload combinations.
   - Evaluate the effectiveness of the attack in uncovering vulnerabilities or harvesting useful data.
   - Document findings, including successful payload combinations and any vulnerabilities discovered.
6. Iterating and Refining:
   - Iterate through different attack configurations and payloads to comprehensively test the application's security.
   - Refine attack parameters based on observed results and feedback to maximize effectiveness.
   - Continuously assess and mitigate identified vulnerabilities to enhance the overall security posture of the application.

RESULTS

- The Proxy tool facilitated the interception and manipulation of HTTP requests, enabling the identification of potential vulnerabilities such as input validation failures and authentication bypasses.
- Utilizing the Repeater tool allowed for iterative testing of modified requests, leading to the observation of changes in the application's behavior and the identification of exploitable weaknesses.
- With the Sequencer tool, the randomness and quality of tokens used within the application were analyzed, providing valuable insights into the effectiveness of token generation mechanisms and the potential for token prediction attacks.

- The Comparer tool facilitated the comparison of data items, enabling the detection of subtle differences that may indicate security vulnerabilities or misconfigurations within the application.
- Finally, the Intruder tool empowered the execution of highly customizable and automated attacks, uncovering input-based vulnerabilities, performing brute-force attacks, and enumerating valid identifiers.

CONCLUSION

We delved into the Community Edition of Burp, demonstrating its vital features such as the proxy, along with a comprehensive history log. Among its essential tools we explored the Repeater, Intruder, Sequencer, and Comparer, each serving crucial roles in the assessment process.

Thus, we found out why Burp Suite, with its efficient identification and mitigation of security risks, is considered an excellent resource for conducting Application Security assessments.

References:

https://portswigger.net/burp/documentation/