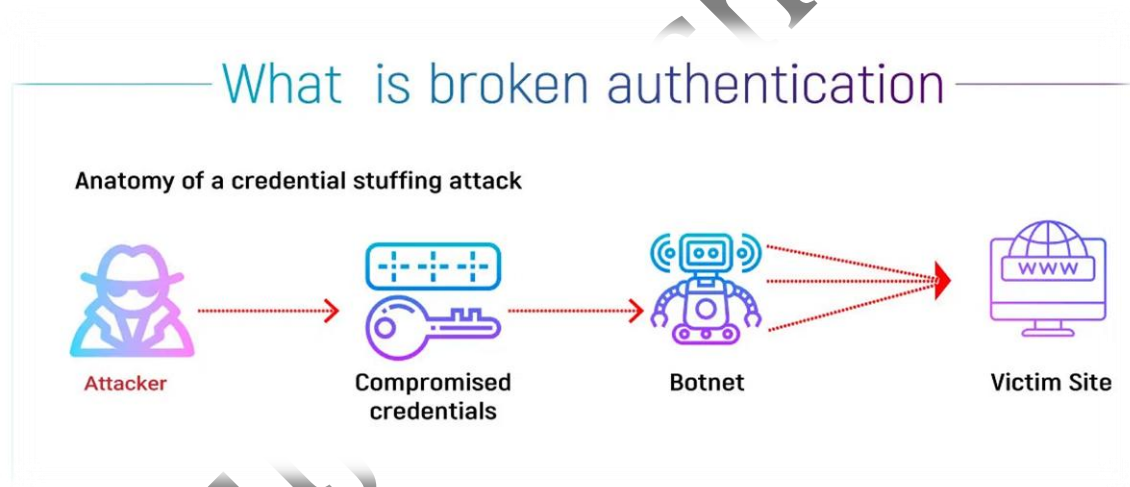


Report on Broken Authentication Tools in OWASP WebGoat and Burp Suite



Anuja Panchariya

Report on Broken Authentication Tools in OWASP WebGoat and Burp Suite

This comprehensive report delves into the issue of broken authentication, a common security vulnerability. It examines how tools like OWASP WebGoat and Burp Suite can be used to identify and demonstrate these vulnerabilities. Broken authentication weaknesses allow attackers to bypass authentication controls, potentially leading to unauthorized access and account takeovers.

Table of Contents

1. Introduction to Broken Authentication
2. Types of Broken Authentication Vulnerabilities
3. OWASP WebGoat and Authentication Testing
4. Using Burp Suite to Detect Authentication Issues
5. Case Studies & Real-World Examples

1. Introduction to Broken Authentication

Broken authentication refers to flaws in the authentication process that allow unauthorized access to resources. These vulnerabilities often stem from weak credentials, poor session management, and insufficient protection of authentication tokens. This section explores the importance of addressing broken authentication in modern web applications.

2. Types of Broken Authentication Vulnerabilities

Broken authentication can manifest in various ways, including:

- **Credential Stuffing:** Attackers use lists of leaked credentials to gain access to accounts.
- **Brute Force Attacks:** Automated scripts repeatedly try different password combinations.
- **Session Fixation:** Attackers set or fix a session ID before the victim authenticates, then use that session to access the account.
- **Improper Logout Management:** Lack of proper logout mechanisms can lead to unauthorized access through persistent sessions.

3. OWASP WebGoat and Authentication Testing

OWASP WebGoat is an interactive training platform that includes lessons on various security vulnerabilities. Its modules on authentication vulnerabilities offer hands-on experience with broken authentication scenarios, such as:

- Password Reset Module: Demonstrates how weak password recovery can lead to unauthorized access.
- Insecure Password Storage: Illustrates the risks of storing passwords in encrypted.

Through these exercises, WebGoat highlights common misconfigurations and weaknesses in authentication mechanisms.

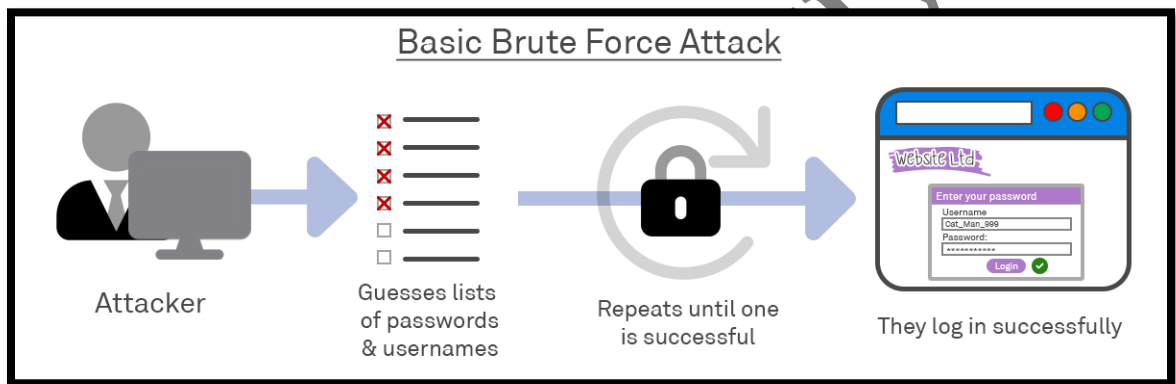


Fig. Basic Brute Force Attack

4. Using Burp Suite to Detect Authentication Issues

Burp Suite is a versatile web application security testing tool widely used for identifying authentication issues. Key features of Burp Suite include:

- Intruder: For brute-force and credential stuffing attacks.
- Repeater: Allows testers to modify and resend requests, testing for session management flaws.
- Sequencer: Analyzes session token randomness and detects weak token algorithms.

Burp Suite's suite of tools makes it particularly effective in pinpointing flaws in login systems and session management.

5. Case Studies & Real-World Examples

To illustrate the impact of broken authentication, consider these real-world cases:

- Case Study: XYZ Corp: An attacker leveraged a weak password reset mechanism to access sensitive user data.
- Session Hijacking Example: In a social media platform, session tokens were exposed over an insecure network, allowing attackers to impersonate users.

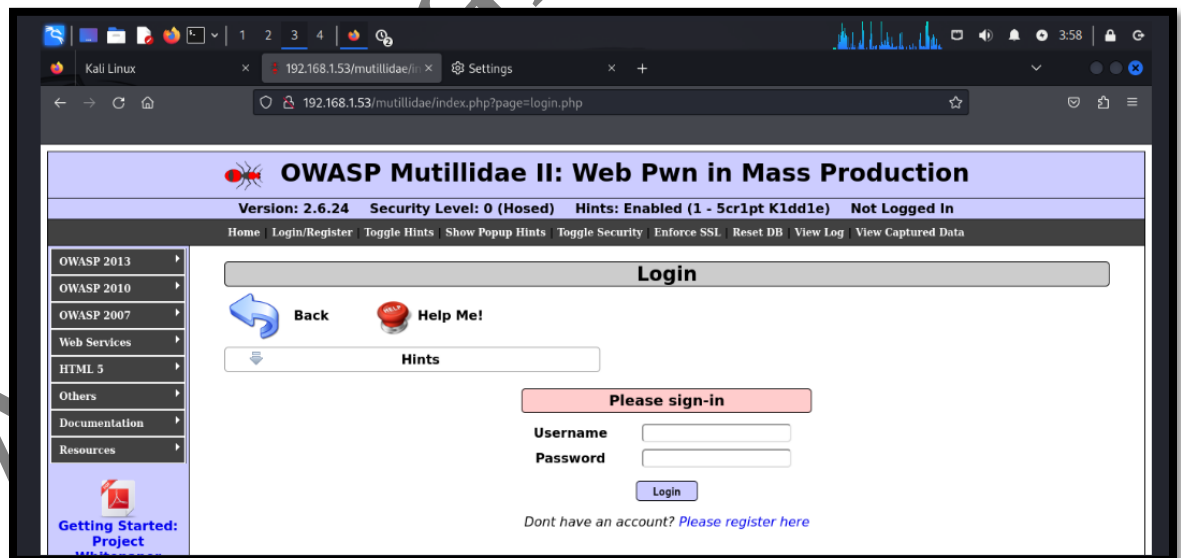
These examples underscore the potential damage that broken authentication vulnerabilities can cause if left unaddressed.

Steps :

1. Broken Authentication Via Cookies

Go to Broken Authentication And Session Management > Authentication Bypass>Via Cookies

- Create Account



- Let's create account as Root

Resources

Getting Started: Project Whitepaper

Release Announcements

Video Tutorials

Please choose your username, password and signature

Username

Password [Password Generator](#)

Confirm Password

Signature

Create Account

- Account created for root

OWASP Mutillidae II: Web Pwn in Mass Production

Version: 2.6.24 Security Level: 0 (Hosed) Hints: Enabled (1 - 5cr1pt K1dd1e) Not Logged In

Home Login/Register Toggle Hints Show Popup Hints Toggle Security Enforce SSL Reset DB View Log View Captured Data

Register for an Account

Back Help Me!

Hints

Account created for root. 1 rows inserted.

Switch to RESTful Web Service Version of this Page

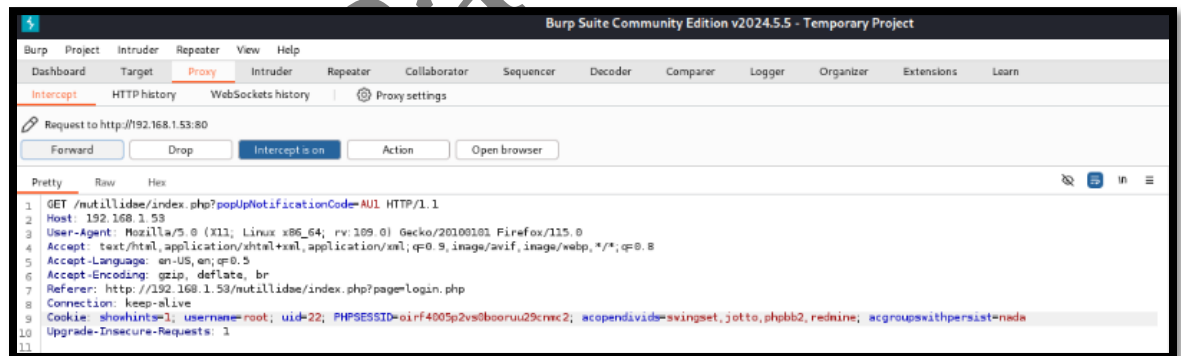
Please choose your username, password and signature

Username

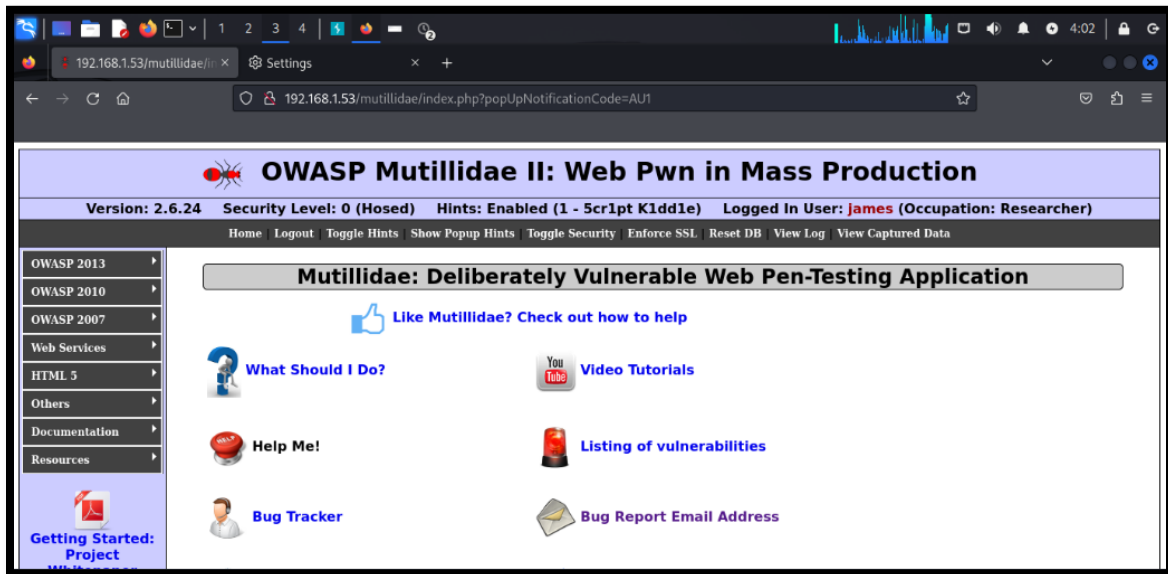
- User logged in as Root



- Intercept and change uid to 22

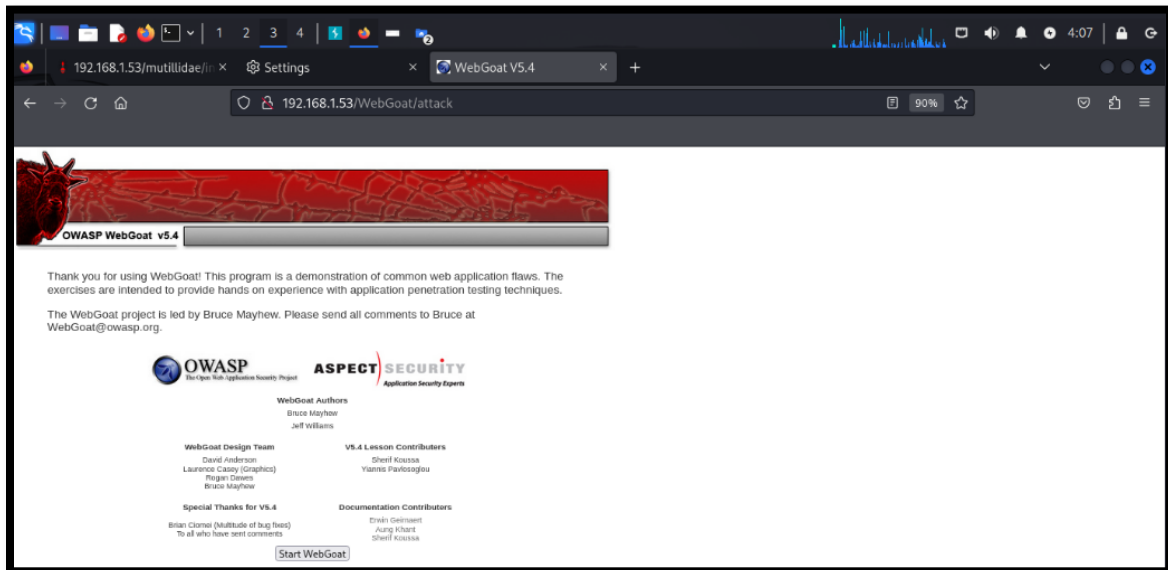


- After changing UID INTO 22 logged in as james

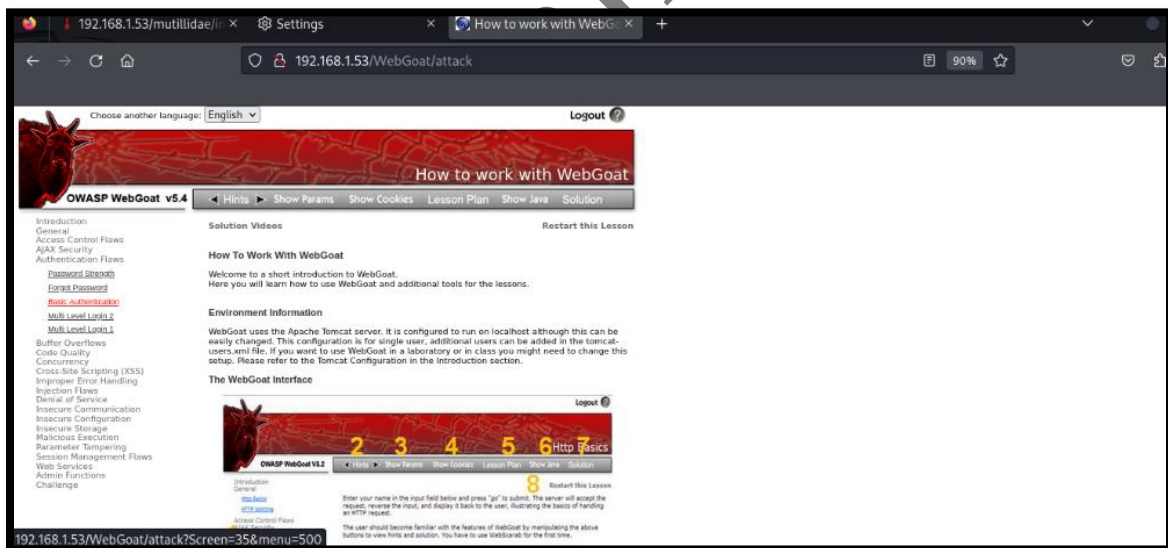


2. Basic Authentication In HTTP Request

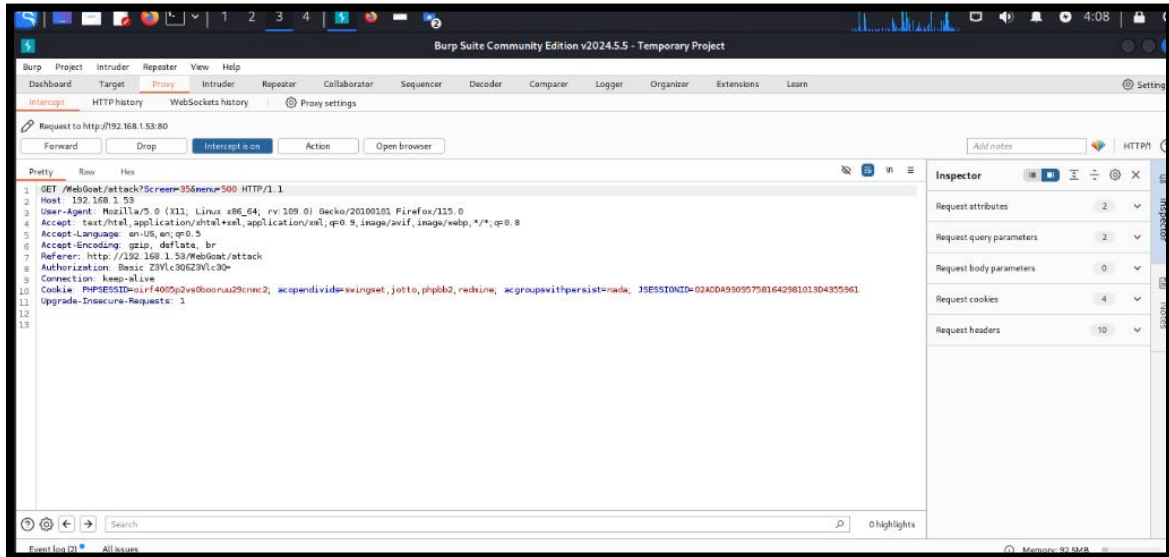
- Go to OWASP WebGoat > Authentication flaws > Basic Authentication



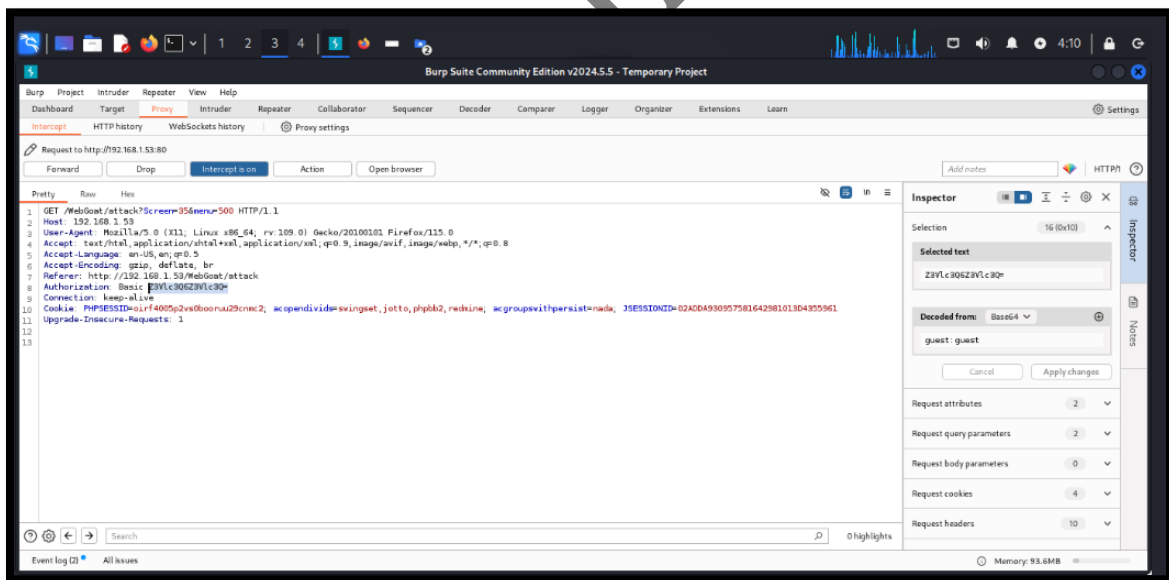
- Go To The Basic Authentication



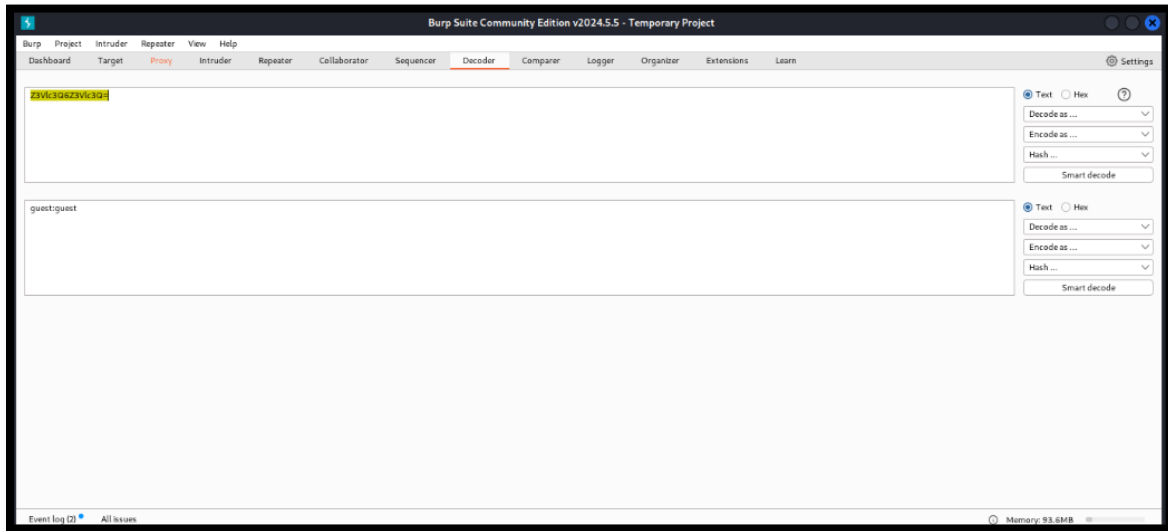
- Turn on the Intercept



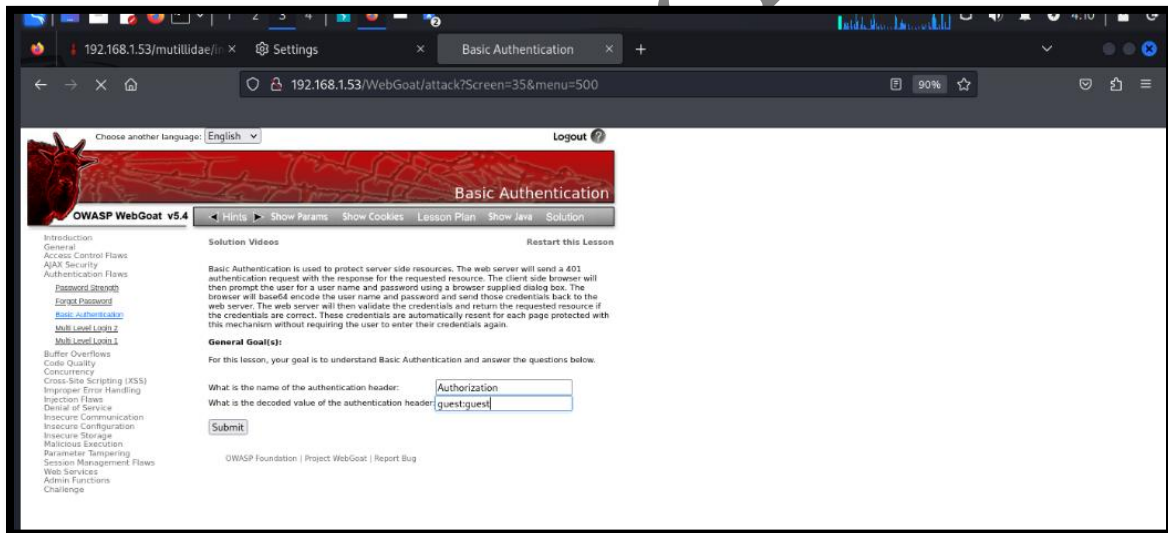
- Decode the Authorization key



- Decoded key is here, guest:guest as username and password

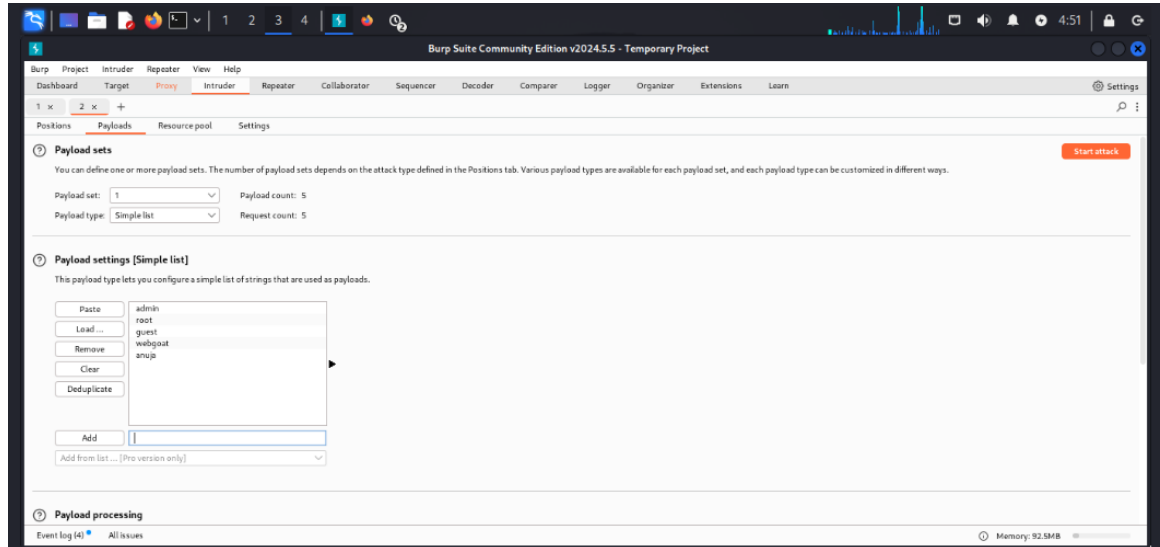


- This decoded key is paste on user and password field

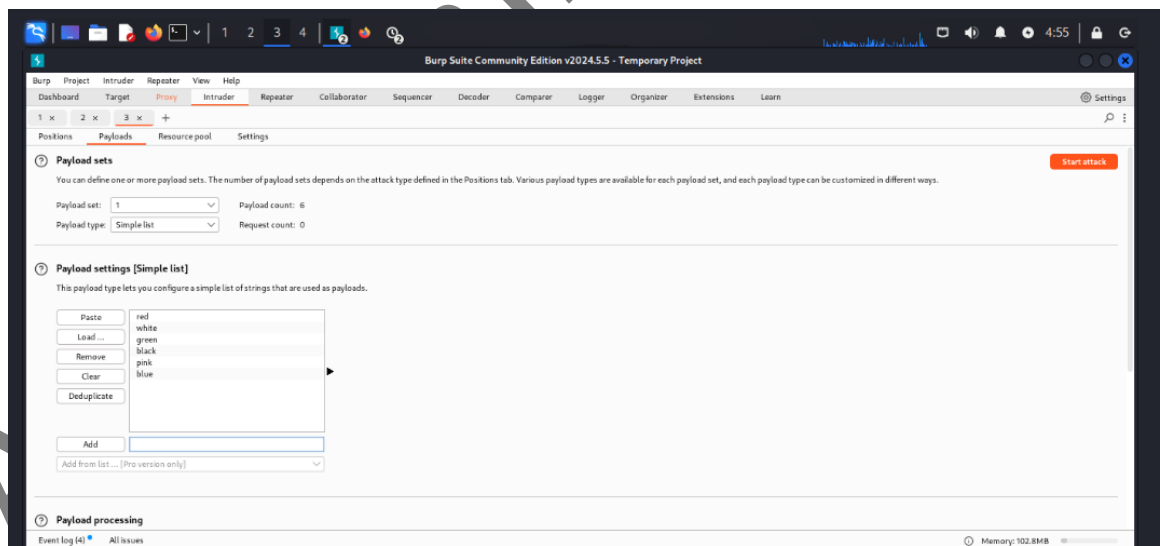


3. Forgot Password

- Go to authentication flaws> forgot Password , logged in as wrong then intercept the request add payloads . Choose attack type to cluster bomb



- Add passwords and perform grep match



- Now , you can access the account

Settings x Forgot Password x New Tab x +

192.168.1.53/WebGoat/attack?Screen=64&menu=500 90%

Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec

Choose another language: English Logout

OWASP WebGoat v5.4

Forgot Password

Introduction
General
Access Control Flaws
AJAX Security
Authentication Flaws
Forgot Password
Basic Authentication
JWT Authentication
Multi-Level Login
Buffer Overflows
Code Quality
Concurrency
Cross-Site Scripting (XSS)
Integer Error Handling
Injection Flaws
Denial of Service
Insecure Communication
Insecure Configuration
Insecure Storage
Malicious Execution
Parameter Tampering
Session Management Flaws
Web Services
Admin Functions
Challenge

Hints Show Params Show Cookies Lesson Plan Show Java Solution

Solution Videos Restart this Lesson

Web applications frequently provide their users the ability to retrieve a forgotten password. Unfortunately, many web applications fail to implement the mechanism properly. The information required to verify the identity of the user is often overly simplistic.

General Goal(s):
Users can retrieve their password if they can answer the secret question properly. There is no lock-out mechanism on this 'Forgot Password' page. Your username is 'webgoat' and your favorite color is 'red'. The goal is to retrieve the password of another user.

*** Congratulations. You have successfully completed this lesson.**

Webgoat Password Recovery
For security reasons, please change your password immediately.

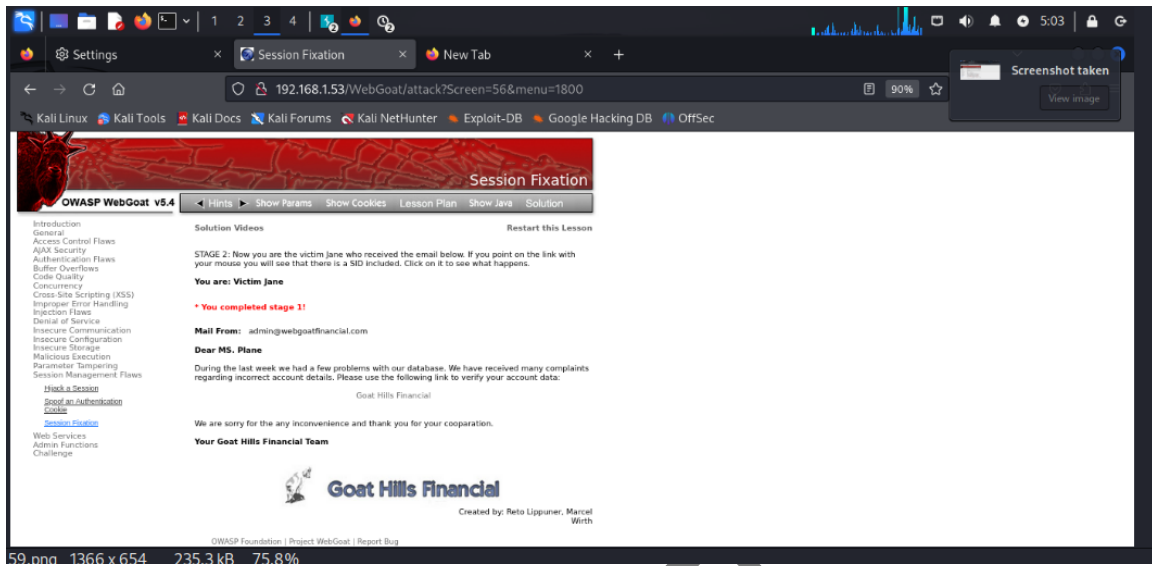
Results:
Username: admin
Color: green
Password: 2275\$tarB0dm3

ASPECT SECURITY
Application Security Experts

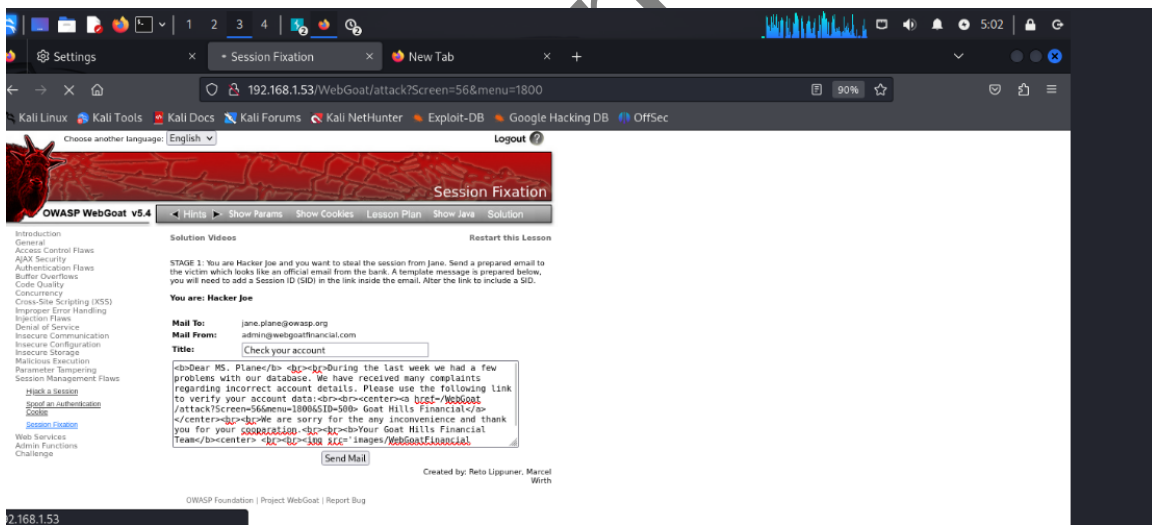
OWASP Foundation | Project WebGoat | Report Bug

4. Session Fixation

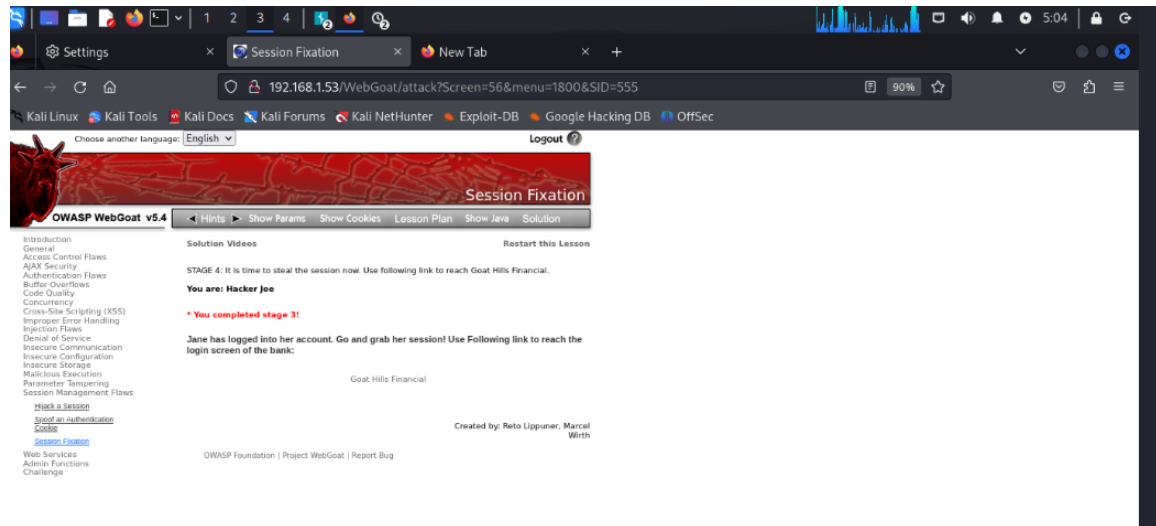
- Go to session management flaws>session fixation



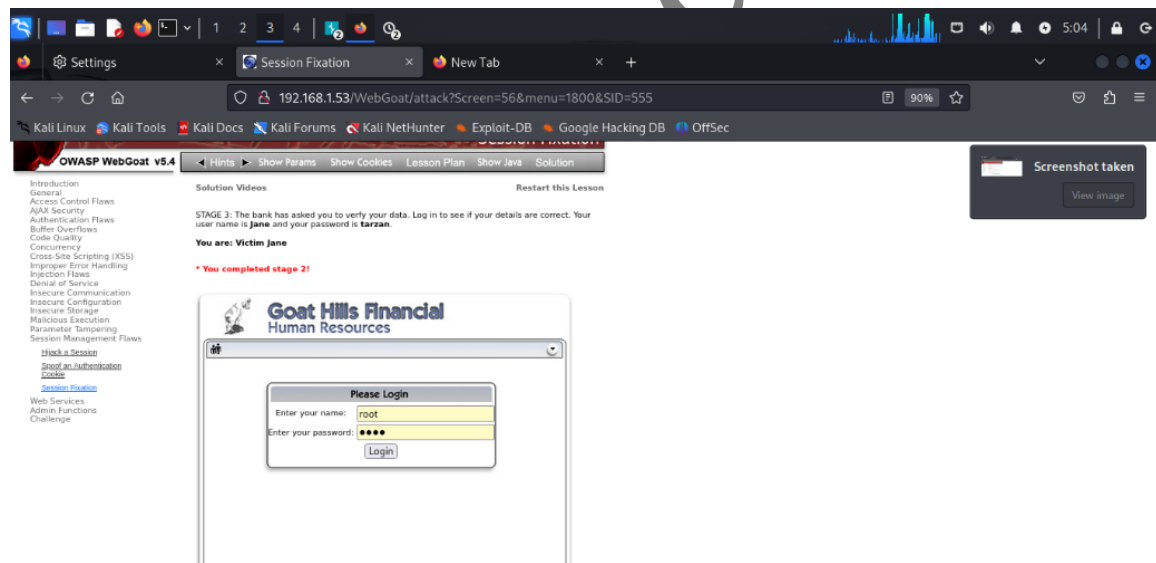
- Change webgoat to WebGoat and send mail



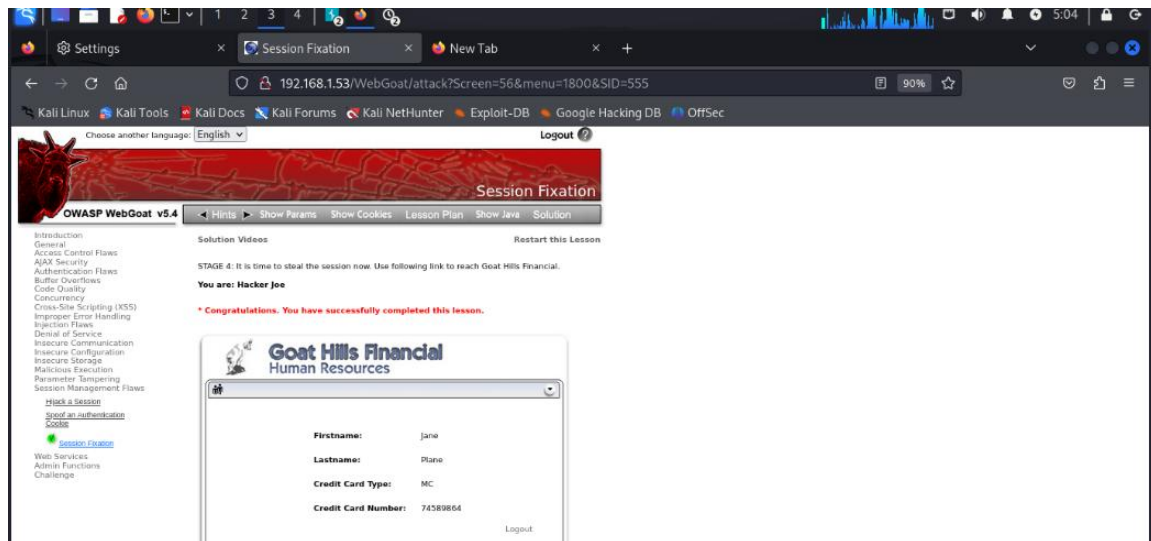
- Click on the link



- Log in



- Log in as victim now , you can access the data



Conclusion

Broken authentication vulnerabilities pose significant risks to web applications and user data. Using tools like WebGoat and Burp Suite, developers and testers can identify and address these vulnerabilities effectively. By understanding common authentication flaws and implementing advanced security measures, organizations can safeguard their applications against unauthorized access.