A

Project

On

Brute Force Sleuthing

For the partial fulfilment of the Course

Linux Shell Programming (CSET-213/CBCA221)

Academic Year: 2024-24 (Odd Semester)

Submitted by

Harshit Maan (E23BCAU0049)

Shashank Chaudhary (E23BCAU0046)

Under The Supervision of

Dr Vimal Kumar/Dr. Debjani Ghosh/Dr Nitin Shukla

School of Computer Science Engineering & Technology

Bennett University (The Times Group), Greater Noida, UP-201310, India

SCHOOL OF COMPUTER SCIENCE ENGINEERING & TECHNOLOGY

BENNETT UNIVERSITY (THE TIMES GROUP),

GREATER NOIDA, UP-201310, INDIA

November 2024

Contents

Topic- Penetration Testing Using a Brute Force Attack

Introduction

In today's age of digital gadgets, cybersecurity threats continue to evolve and increase, brute force attacks being one of them and still remaining a significant concern for organizations. Our project aims to demonstrate a brute force attack on a vulnerable web application, A-Z Edu, to highlight the importance of robust, password policies and security measures that users (not just users) should take while creating an account.

Background

A brute force attack in simple words is a trial-and-error method used to systematically submit passwords or passphrases (string of words) to a system until the correct one is found. This technique is used to compromise accounts and gain unauthorized access to sensitive information.

Project Scope

Target Application: A-Z Courses (sample website)
Vulnerability Exploited: Weak password policy
Tools and Techniques: Burp Suite, Sniper attack technique, set of common and custom words as a wordlist

Addressed the Problem: Targeted Applications

There have been various cases of user accounts being compromised because of weak password policies, or users keeping weak passwords

on their own maybe so that they don't forget or be it any reason, one such incident that happened was with 'Dunkin' donuts'. In 2015, a famous incident had happened, involving the use of brute force at Dunkin' Donuts' digital customer accounts, these accounts were targeted by hackers who used a leaked list of previously stolen credential information and ran brute force algorithms. They gained access to 19,715 user accounts for the customer loyalty application and stole tens of thousands of dollars of rewards cash. And DD had to pay over half a million in penalties ($650,000).

Project Objectives

Objectives of our project can be described in the following manner:

1) **Primary Objective:** To successfully compromise an account of A-Z Edu. By using a brute force attack through burp suite.
2) **Secondary Objectives:**
    - To identify vulnerabilities in the website's security measures, specifically related to password strength and authentication mechanisms.
    - To demonstrate the effectiveness of brute force attacks and the importance of implementing strong security controls.
    - To analyze the time taken to successfully compromise the account and the impact of different attack techniques.
    - To finally provide actionable recommendations to improve the website's security posture, from whatever we have learnt about the security measures through the attack(/s)

Software Requirement Specification (SRS) & Playbook Design:

1) **Functional Requirements:**
    - *Tool configuration:* Ability to configure Burp Suite for intercepting HTTP requests and responses; Ability to set up the Intruder module with the appropriate attack type (Sniper) and payload options (wordlists).
    - *Attack Execution:* Ability to launch the brute force attack and monitor its progress; Ability to pause, resume, or stop the attack as needed; Ability to log attack results, including successful login attempts and failed attempts.
    - *Result Analysis:* Ability to analyse the attack results and identify any vulnerabilities.
2) **Non-Functional Requirements**:
    - *Performance:* The attack should be efficient and complete within a reasonable timeframe.
    - *Security*: The tool should be secure and should not compromise the target system beyond the scope of the attack.
    - *Usability*: The tool should be easy to use and understand, with clear instructions and a user-friendly interface.

Playbook Design:

Phase 1: Reconnaissance

1) *Identify Target:* Determine the target website and its login page.

2) *Gather Information***:** Collect information about the website's technology stack, security measures, and potential vulnerabilities.

Phase 2: Tool Configuration

1) *Install Burp Suite***:** Install and launch Burp Suite.
2) *Proxy Configuration***:** Configure Burp Suite as a proxy to intercept HTTP traffic.
3) *Intruder Setup***:** Set up the Intruder module with the following parameters:
    - Attack Type: Sniper
    - Payload Type: Wordlist
    - Payload Positions: Target the username and password fields.
    - Wordlists: Load the prepared wordlists.

Phase 3: Attack Execution

1) *Launch Attack**:*** Start the brute force attack.
2) *Monitor Progress:* Monitor the attack's progress and log successful login attempts.
3) *Analyze Results**:*** Review the attack logs to identify any successful login attempts or other vulnerabilities.

Phase 4: Post-Attack Analysis

1) *Security Implications:* Assess the security implications of the successful attack.
2) *Recommendations:* Provide recommendations to improve the target website's security, such as implementing stronger password policies, enabling multi-factor authentication, and using web application firewalls.

Playbook Design: Preventing Brute Force Attacks

1) *Strong Password Policies:*

- Enforce strong password policies, including:
  - Minimum password length
  - Required character complexity (uppercase, lowercase, numbers, special characters)
  - Regular password expiration
  - Password complexity checks

2) *Multi-Factor Authentication (MFA):*

- Implement MFA to add an extra layer of security.
- Use methods like time-based one-time passwords (TOTP), biometric authentication, or hardware tokens.

3) *Account Lockout:*

- Lock accounts after a certain number of failed login attempts.
- Consider implementing IP-based rate limiting to further mitigate attacks.

4) *Web Application Firewall (WAF):*

- Deploy a WAF to filter and block malicious traffic, including brute force attempts.
- Configure the WAF to detect and mitigate common attack patterns.

5) *Security Headers:*

- Implement security headers like Strict-Transport-Security (HSTS) to enforce HTTPS and X-Frame-Options to prevent clickjacking.
- Use Content-Security-Policy to restrict the resources that can be loaded on your website.

6) *Regular Security Audits and Penetration Testing:*

- Conduct regular security audits and penetration tests to identify and address vulnerabilities.

Implementation

Tools and Technologies:

1) *Burp Suite:* A comprehensive web security testing tool used to intercept and modify HTTP requests and responses.
2) *Python:* A versatile programming language used for flasking

Steps Involved:

1) *Target Identification:*
    - Identify the target application or website for the brute force attack.
    - Analyze the login page to determine the necessary parameters (username, password, etc.).
2) *Burp Suite Configuration:*
    - Set up Burp Suite as a proxy to intercept HTTP traffic.
    - Configure the Intruder tool with the target URL, attack type (e.g., Sniper), and wordlists.
3) *Launch the Attack:*
    - Initiate the brute force attack using the Intruder tool.
    - Monitor the progress of the attack and adjust parameters as needed.
4) *Implement Security Measures:*
    - Based on the findings of the attack, implement appropriate security measures to prevent future attacks, such as:
        - Strong password policies
        - Multi-factor authentication
        - Web application firewalls
        - Rate limiting
        - Security headers

- Regular security audits and penetration testing

Resources Used

Software:

- *Burp Suite Professional:* A comprehensive web security testing tool used to conduct the brute force attack.
- *Python:* versatile programming language, here used for flask app and adding templates
- *Text Editor:* A text editor like Visual Studio Code, Notepad++, or Sublime Text for writing and editing scripts and reports**.**

Hardware:

- Computer system with sufficient processing power and memory
- Reliable internet connection for processing HTTP requests

Phases and Milestones

Phase 1: Project Planning and Setup

- **Milestone 1.1:** Define project scope and objectives.
- **Milestone 1.2:** Set up the development environment (Burp Suite, Python, etc.).
- **Milestone 1.3:** Identify the target application and its vulnerabilities.

Phase 2: Brute Force Attack Execution

- **Milestone 2.1:** Configure Burp Suite for the attack.

- **Milestone 2.2:** Launch the Intruder attack with appropriate payloads.
- **Milestone 2.3:** Save the attack results in a CSV file.

Phase 3: Automation and Optimization (to be worked upon and added later on)

- **Milestone 3.1:** Develop a Burp Suite extension to automate the attack process and data extraction.
- **Milestone 3.2:** Implement automated report generation using Python scripting.
- **Milestone 3.3:** Optimize the attack process and data analysis for efficiency.

References

BurpSuite CE: https://portswigger.net/burp/communitydownload

Dunkin' Donuts 2015 Incident: https://informer.io/resources/dunkin-donuts-data-breach

A-Z education: https://github.com/akshatmiglani/Brute-force-for-login-bypass-on-a-local-website/blob/main/app.py

Understanding BurpSuite: https://www.youtube.com/watch?v=hY_gzrTMn3U&list=PLwO5-rumi8A7TVRzfOD4OHabwJ0v1ZA81&index=1

LinkedIn Post about our project: https://www.linkedin.com/posts/harshit-maan-187936249_3rd-semester-cyber-security-project-name-activity-7268198101776015363-cVHD?utm_source=share&utm_medium=member_desktop