LONDON METROPOLITAN UNIVERSITY

Islington college
(इस्लिङ्टन कलेज)

Module Code & Module Title

CC5009NI Cyber Security in Computing

Assessment Weightage & Type

60% Group Coursework 02

Year and Semester

2024 -25 Autumn Semester

Student Name: Rebika Shrestha London Met ID: 23056194

Student Name: LilaRaj Dura London Met ID: 23056178

Student Name: Shreya Bastola London Met ID: 23056226

Assignment Due Date: 2025/04/22

Assignment Submission Date:2025/04/22

Word Count (Where Required):7931

I confirm that I understand my coursework needs to be submitted online via MST under the relevant module page before the deadline for my assignment to be accepted and marked. I am fully aware that late submissions will be treated as non-submission and a mark of zero will be awarded.

Acknowledgement

We would like to express my sincere gratitude to everyone who supported us throughout this coursework . First and foremost, I would like to thank my Module teacher, Mr. Samrid Budhathoki for their guidance, patience, and invaluable feedback. Their expertise and encouragement made this coursework possible. We also wish to acknowledge the contribution of the online cybersecurity communities, which provided us with the resources and practical insights needed to understand and demonstrate brute force attacks in an ethical manner. Lastly, We would like to thank my friends and family for their constant support and understanding during the completion of this project. Without their encouragement, this journey would have been much more challenging.

Abstract

Brute force attacks remain one of the most common and persistent threats targeting IT systems today. This report takes an in depth look at what brute force attacks are, beginning with basics of their mechanisms, classifications, and common techniques and why they are such a significant security concern. It talks into real-world implications these attacks pose to IT infrastructure, followed by a demonstration of a brute force attack using ethical hacking tools within a controlled ethical environment. Following the Penetration Testing Execution Standard (PTES) framework, the report highlights the methodologies used for exploration, exploitation, and mitigation, offering detailed understanding of ethical practices in cybersecurity. Additionally, the report talks about defensive strategies and technologies that can be done to safeguard systems against brute force threats.

Table of Contents

List of figures

1. Introduction

In this 21st century, everything requires internet. From our normal to the most confidential data, everything is kept in the internet. Internet is used widely all around the world. The internet is like a double-edged sword, while it provides numerous benefits, it also comes with significant risks and challenges. one of the major threats being cyberattacks. A cyberattack is an attempt to steal, alter, destroy, disrupt, or compromise information and systems found within computer networks (Staff, 2025). The attacker can be an individual person or a group of people aiming to gain unauthorized access to the computer/ laptop and steal or modify the data for financial gain, espionage, or other malicious purposes. They often use malware, viruses, or other malicious software to break into systems, steal sensitive information or cause damage. Such attacks can lead to huge potential loss to the victims and can cause cybercrimes like identity theft, financial losses, and data breaches. These attacks are classified into two types:

i) **Active Attacks:** In this type of attack, the victim becomes aware of the attack immediately as the attacker modifies or disrupts the data.

ii) **Passive Attacks:** In this type of attack, the attacker silently monitors or collects data without the victim's knowledge so the victim is unaware of the attack.

These cyberattacks comes in many forms, each with different techniques and goals. One of the most common and dangerous type is brute force attack. A brute-force attack is a hacking method used to login credentials, encryption keys, and hidden URLs. As the name implies, brute force attack uses brute force techniques in the form of endless login attempts to gain unauthorized access to private accounts, sensitive files, organizations' networks, and other password-protected online resources (Scott, 2024). In a brute-force attack, the attacker tries every possible password or passphrases hoping to eventually guess the password correctly. The attacker checks all the possible passwords, letters and passphrases until the correct one is found. Sometimes instead of guessing the password directly, the attacker targets the key. This is called as exhaustive key search, a cryptanalytic attack that attempts to decrypt any encrypted data by trying every possible key. While brute-force attacks can be very effective for cracking short or weak passwords, they become much less practical as the length and complexity of the password increases. So, for longer passwords, other methods such as a dictionary attacks are used because a brute-force search for long username and password takes too long to crack. The effectiveness of brute-force attack depends upon how

complex and lengthy the target password is as the longer passwords with more characters are more exponentially increase the time required to crack them. (Swathi, 2022)

Figure 1:Brute force attack working.

1.1. Technical terminologies

To fully understand how brute-force attacks are conducted, especially within the framework of ethical hacking and penetration testing, it's impo3rtant to first become familiar with some of the key technical terms used throughout this report. These terminologies form the foundation for explaining the methods, tools, and processes involved in both offensive and defensive cybersecurity practices.

i. Authentication

Authentication is the process of verifying that a user or device is truly who or what it claims to be before giving access to a system or resource. It ensures that only authorized persons can access the system, typically by requiring credentials like passwords, biometrics, or security tokens. It is like the second step in a three-step process: identification (who you are?), authentication (prove it), and authorization (if you have permission?). For example, when you type in a username (identification), the system also asks for a password or fingerprint (authentication) to confirm it's really you. Once you're verified, the system then decides what you're allowed to do (authorization). Hence, authentication is an important step to keep digital systems safe by making sure only the authenticated users get in. (Heyman, 2024)

ii. Brute Force Attack

The terminology "brute-force" comes from the method of using repeated, many attempts to guess login details until the correct one works and access is gained. It’s like using every possible key until one finally unlocks the door. In most of the cases, attackers start by using basic personal information like person's names, birthdays, or hobbies to take guesses. These attacks can be done manually or by using special automated using tools that can try hundreds or even thousands of combinations per second. While brute-force is a simple technique in concept, it can be very effective if systems don’t have strong security or password policies. (Martinez, 2024)

iii. Dictionary Attack

A dictionary attack is a type of brute-force attack where the attacker tries to guess someone's login details using a pre-made list of common passwords, words, or phrases known as a "dictionary" which helps to guess a user's login credentials. Since many users still use simple or predictable passwords like “password123” or “admin.”, this method can surprisingly be effective. Instead of testing every possible character combination, the attacker only tries from the list. Dictionary attacks are faster and are often used if people reuse weak passwords across multiple sites. Cybercriminals often use automated tools for this purpose, especially when targeting systems with poor password security. (O'Sullivan, 2024)

iv. Vulnerability

A vulnerability is a weakness in a computer system, implementation, or configuration that cybercriminals can exploit to gain unauthorized access or perform malicious activity. These weaknesses can exist in any form, it can be in software, hardware, networks, or even human factors like weak passwords. Once an attacker discovers a vulnerability, they can exploit it to install malware, or steal sensitive information such as credentials, financial information, or intellectual property. (Goodman, 2025)

v. Exploitation

Exploitation refers to the act of taking advantage of a system's vulnerabilities or weaknesses to gain unauthorized access. Cybercriminals exploit these weaknesses to steal data, install malware, or disrupt services. This can affect individuals, businesses, and organizations, often leading to financial loss or data breaches. (Raza, 2025)

1.2. Aims and Objectives

The aim of this report is to analyze brute force attacks on IT devices and systems, examining their mechanisms, impact, and real-world examples of their implications. This report will access brute-force attack, demonstrate how brute force attacks are executed, talk about various mitigation techniques, and evaluate their effectiveness in preventing unauthorized access.

The objectives of this report is to:

- To define brute force attacks and explain their role in cybersecurity threats.
- To learn about different types of brute force attacks, including dictionary attacks, credential stuffing, and hybrid attacks.
- To demonstrate the process of executing a brute force attack using ethical hacking tools in a controlled environment.
- To learn the impact of brute force attacks on IT systems, businesses, and individuals.
- To talk about various mitigation techniques, such as account lockouts, CAPTCHA, multi-factor authentication (MFA), and encryption.
- To evaluate the effectiveness of different mitigation strategies by testing their impact on brute force attack attempts

1.3. Report Structure

This report is divided into five main key sections to provide a comprehensive understanding of brute force attacks:

Introduction: The introduction section introduces the concept of brute force attacks and defines its key terminologies such as authentication, brute-force attacks, and exploitation. It also outlines the aims of the report.

Background: The background section explores different types of brute force attacks. This section also explains the Penetration Testing Execution Standard (PTES), a structured methodology used to perform ethical hacking and penetration testing alongside its seven stages and tools used in each section.

Demonstration: The demonstration section provides a practical walkthrough of a brute force attack using tools within a controlled environment. As a part of the demonstration, a formal Terms of Reference (ToR) was prepared to outline the key elements of the evaluation process.

Mitigation: Following this, the report also discusses the mitigation strategies and how to defend against brute force attacks.

Evaluation: The evaluation section focuses on how to defend against brute force attacks, evaluates the effectiveness of mitigation strategies, and discusses real-world applications.

Conclusion and References: Finally, the report concludes with a summary of findings and a list of references used throughout the research.

2. Background

In brute force attack, the attackers use bots to crack password combinations. They have a list of combinations of words and letter from login details and leaked credentials obtained from past data breaches, the attacker manually guesses the credential details, trying their luck for user logins. Guesses may be time-consuming and hard to crack since many attackers may be an impatient person, they take the help of a software or other brute force tools to get multiple combinations for cracking the website. With the help of these tools, these attackers can attempt multiple password combinations ids for login the website and applications. (Christian, 2024)

2.1. Types of Brute-force attack

Brute-force attacks, though simple in concept, it can vary in execution depending on the attacker's strategy and the target's security measures. The following are some of the most common types of brute-force attacks and how they operate.

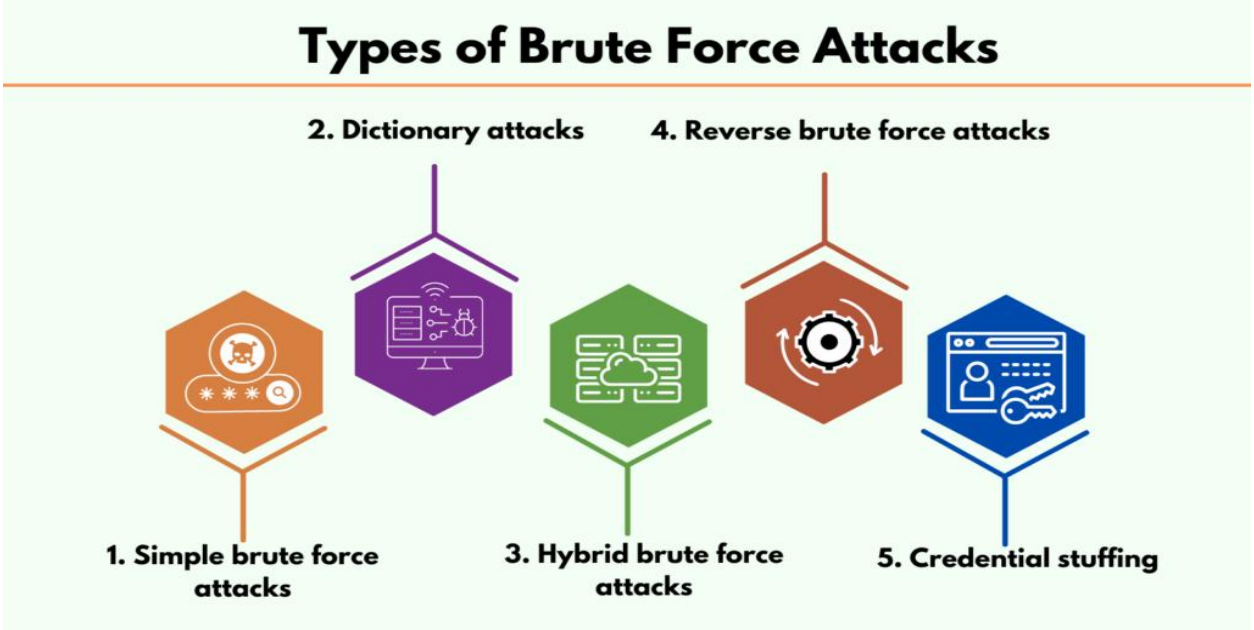

Figure 2:Types of brute force.

i**) Simple brute force attack:** A simple brute force attack known as a way of simply cracking by lucky guesses. Hacker try to guess a user's login credentials manually without any software. This is typically through standard password combinations or personal identification number(PIN)codes. These attacks are a simple method because many people still use weak passwords, such as a "user123" or "12345" or a practice poor password, such as using the same password for the

multiple websites. Passwords can also be guessed by the hackers that do minimal spying, observations work to crack an individual's potential password, such as a name of their favorite sports team. It uses scripts to guess the passwords. (Lenaerts-Bergmans, 2022)

ii**) Dictionary Attack:** A dictionary attack is a password cracking technique that serially tries commonly used words, phrases, and leaked passwords to gain unauthorized access to an account or system. Dictionary attack relies on precompiled wordlists that include frequently used passwords, names, popular phrases, and variations with character substitutions like replacing 'a' with '@'. Modern dictionary attacks use massive databases of leaked credentials from previous breaches, many of which are available on the dark web. Advanced tools like Hashcat tend to enhance these attacks by combining words, adding common prefixes or suffixes (for e.g. 'user123'), and using hybrid methods that blends dictionary-based and brute-force techniques. Because users often choose very predictable passwords, dictionary attacks are highly effective and significantly faster than traditional brute-force methods. (Lenaerts-Bergmans, 2022)

iii) Hybrid Brute Force Attack: A hybrid attack combines brute force and dictionary attacks. Hackers use common words and phrases, then try variations by adding characters, mimicking how users create passwords. Since many people use predictable patterns, this method is faster and more efficient than guessing every possible combination. (Mohammed, Bello Suleiman1 ; Romanus, Robinson2, 2024).*Password length, complexity, hashing algorithm used, and the computational power of the attacker's system algorithms are some of the major things that determine the effectiveness of brute-force attacks. The longer complex passwords increase the number of possible combinations, making brute force attempts significantly slower. Finally, attackers with powerful hardware, like GPUs or distributed systems, can perform billions of hash calculations per second, reducing the time needed to crack weak passwords.

iv) **Reverse Brute Force Attack:** Reverse attack is widely known method where the attacker starts with a known password, either gather from a breach or commonly used, then searches and attempts many usernames until the combination is found. Different from a traditional brute force because they are working a backwards and starting with the known passwords instead of known the usernames. For example, a simple option such as "Password" may be used to brute force a username that goes with it. Unlike traditional brute force attacks, which attempt to crack passwords for a specific username, reverse brute force makes the process by starting with the passwords as

the key to access. Additionally, reverse brute force attacks are difficult to detect, as attackers spread out failed login attempts across multiple accounts to avoid triggering security alerts. (Mohammed, Bello Suleiman1 ; Romanus, Robinson2, 2024)

v) **Credential Stuffing**: Over the years, more than 8.5 billion users and passwords have been leaked, leading to a rise in dictionary attacks and credential stuffing, two common techniques used by cybercriminals to crack passwords. Credential Stuffing is a type of cyberattack in which the attackers use compromised credentials to gain unauthorized access to user accounts. This technique is based on the fact that many people use multiple websites and services with the same username and password combination. As a result, hackers can simply try these credentials on other platforms until they find out one that works. Once an attacker has gained control over an account, they can steal sensitive information, make scams, or carry out various other illegal activities. This attack method poses significant risks to both individuals and organizations, and thus the need for strong passwords, multi-factor authentication and mindful security measures is very important. Credential stuffing can cause extremely damaging impacts on businesses by bringing about a complete set of consequences related to financial stability, customer trust, and compliance with rules. (Larson, 2024)

To conduct brute-force attacks ethically and in safe environments it is necessary to follow a structured and clear methodology. This ensures that the test is not only the legal and safe but also the accuracy and usefulness of the findings. One of the most widely accepted framework used by professionals for such processes is known as the Penetration Testing Execution Standard (PTES). PTES breaks down the process into easy-to-follow set of stages that helps penetration testers plan, execute, and report their activities in a controlled and professional way. This standard is mainly useful when trying to copy real-world attacks, such as brute-force login attempts, in ethical hacking environments.

2.2. The Penetration Testing Execution Standard (PTES)

The Penetration Testing Execution Standard (PTES) is a clear and structured set of rules that is made to guide penetration testing processes and improve the overall quality and consistency of ethical hacking practices. Before PTES was introduced, there were no proper rules for penetration testing, often resulting in inconsistent results, little oversight, and lacked quality. Hence, PTES was introduced in 2009 by a small group of cybersecurity experts who wanted to fix6 these issues.

Today, this has expanded to a team of around 20 senior information security practitioners, mostly based in the United States. The idea for creation of PTES was from discussions among its founding members about the growing concern over the effectiveness and value of penetration testing in the industry. This methodology gives both businesses and security service providers with a common language, defined expectations, and a clearly outlined scope for conducting penetration tests. The PTES standard mainly targets two key audiences: businesses that require penetration testing services and the providers who deliver them. For businesses, PTES sets a clear understanding for business to know what to expect during a penetration test, and ensures that they receive consistent, measurable, and actionable results. For service providers, it offers a structured step-by-step approach to conducting penetration tests that covers everything from the initial planning and information gathering phases to execution, reporting, and final deliverables. By doing this, PTES helps build trust between the testers and clients, improves communication, transparency, ultimately raising the overall reliability and effectiveness of penetration testing in cybersecurity. (Sahoo, 2025)

To carry out brute force testing ethically and responsibly, we followed the Penetration Testing Execution Standard (PTES). It has seven stages, and we used each one to plan and demonstrate our attack in a safe, structured way.

Figure 3: Stages of PTES.

i. Pre-engagement Interactions

Before starting any kind of ethical hacking or penetration testing, it is important to have a clear agreement between the tester and the organization. This step is called Pre-Engagement Interactions in the PTES framework. It's where both sides talk about what the test is for, what parts of the system will be tested, and what the rules are. For example, in the following brute force attack demonstration, this would include agreeing that only a specific login page in a test environment will be targeted, and not real user accounts or sensitive data. The goals of the test should mainly focus on checking the system's security, and also make sure that everything is done legally. The scope of the test like what tools are allowed, how long the test will run, and which systems or features are "off-limits." are set This planning stage helps make sure the test is safe, legal, and useful for finding real security issues. (Ridwan, 2024)

A Terms of Reference (TOR) document was also prepared as part of this phase. It outlined the assignment title, attack type (VAPT), the 30-day testing duration, the client (DVWA), and the pen testers involved. The TOR also included the background, study objective, scope of evaluation, expected deliverables, intellectual property rights, and evaluation criteria. Including this helped ensure that everyone involved had a shared understanding of roles, goals, and responsibilities. This phase ensured the test remained safe, legal, and productive for identifying real-world vulnerabilities.

ii. Intelligence Gathering

In the Intelligence Gathering phase, relevant information is collected about the system or the target without actually interacting with it directly. This step is also called OSINT (Open-Source Intelligence) and it involves using publicly accessible sources like search engines, social media, domain lookups, employee profiles and other online resources to understand the target. In the case of brute force attack demonstration, this step is important for finding out potential usernames, login portals, or weakly secured systems that could be exploited later. PTES breaks this into three levels of intelligence gathering: Level 1 focuses on just the basics, like what security tools are in place. Level 2 digs deeper, looking into organization's security handles and operational practices. Level 3 is the most advanced and involves comprehensive research tries to uncover hidden details like tech partnerships or exposed employee credentials. All of this findings help in building a plan for which systems might be vulnerable to brute force attacks later on. (Ridwan, 2024)

Kali Linux is a special operating system designed for testing the security of computer systems and networks (1k, 2024).Metasploitable 2 is a type of virtual machine based on Linux and is intentionally made vulnerable so that users can practice penetration testing tools like Metasploit. Nmap (Network Mapper) is an open-source command-line tool which is primarily used for network discovery and security auditing. It allows network administrators to scan IP addresses and ports, detect devices, and identify vulnerabilities (Shivanandhan, 2020). In the Intelligence Gathering phase of the PTES framework, the main goal is to collect essential information about the target system. Nmap was used to perform network scanning and service enumeration on the Metasploitable2 machine, identifying open ports and running services. The test environment was hosted and operated using Kali Linux, which provided all necessary security tools and terminal access. Crunch is a highly flexible command-line tool used to create custom wordlists for brute-force attacks or password recovery. It is commonly used by penetration testers and ethical hackers to create targeted password lists based on specific rules or patterns. To prepare for the brute force attack, the Crunch tool was utilized to generate a custom password wordlist tailored to the login form in the target application. This wordlist would later be used to systematically guess user credentials during the attack. Together, these tools enabled a strong foundation for understanding the target system and planning the attack path.

iii. Threat Modeling

Threat Modeling is the third phase of the Penetration Testing Execution Standard (PTES) and plays an important role in shaping the attack strategy. After gathering enough information about the target system, penetration testers use threat modeling to identify which assets are most valuable and vulnerable. This helps them determine how an attacker might exploit those weak points. The process begins by collecting detailed documents such as network architecture, system inventories, and access control policies. Then after the categorization they are sorted and prioritized based on their importance and risk exposure. Once the critical assets are defined, potential threats are categorized—ranging from external attackers to insider threats such as malicious insiders, cybercriminal groups, automated bots or software vulnerabilities. Finally, these threats are mapped to the identifying assets they might impact, allowing testers to predict which areas are most at risk. This stage is especially important for brute-force attack simulations, as it helps narrow down targets like login portals, admin panels, or authentication systems that could be vulnerable to

password-guessing attempts providing effective strategy for planning responsible and targeted testing. (Ridwan, 2024)

iv. Vulnerability Analysis

Once threat modeling is complete, now actual weakness in the system should be identified. In this phase of the PTES standard, the penetration tester closely examines the systems to find weaknesses that could be targeted in a real attack. For brute-force attacks, this stage is especially important because it helps the ethical hacker locate login pages, misconfigured authentication systems, or exposed services that could be vulnerable to password-guessing attempts. PTES outlines two approaches during vulnerability analysis:

Passive analysis: It involves minimal interaction with the target system. This may include monitoring network traffic or analyzing metadata to discover basic weaknesses without actively looking at the system. In the context of brute-force attacks, this might include checking exposed login endpoints, analyzing error messages, or reviewing metadata to understand how authentication is handled.

Active analysis: It is more intrusive and hands-on as it involves directly interacting with the system. For example, scanning ports to discover services like SSH or FTP, or using tools like Hydra in a test environment to simulate brute-force login attempts on purposely vulnerable platforms like DVWA (Damn Vulnerable Web Application). (Ridwan, 2024)

By carefully identifying login-related weaknesses using both passive and active techniques, the penetration tester can decide the best way to ethically demonstrate a brute-force attack during the exploitation phase. Nessus Scanner is a powerful vulnerability scanning tool which was developed to perform vulnerability assessment and penetration testing using Common Vulnerability and Exposure architecture For Vulnerability Analysis phase, the Nessus Vulnerability Scanner was used to identify known vulnerabilities within the target system. Nessus helped provide a detailed report of potential security flaws, including misconfigurations and outdated services, which helped in having a clear view of vulnerable or exploitable areas in the system. This step helped in verifying the existence of the brute force vulnerability and set the stage for the actual exploitation process.

v. Exploitation

This is the phase where all the preparation comes together and execution of the attack takes place. All the previously gathered vulnerabilities is strategically used to gain unauthorized access. In the context of brute-force testing, the tools like Hydra or DVWA are used to target login portals attempting combinations of usernames and passwords. This step aims to mimic a real-world attacker behavior within the ethical boundaries, testing the flexibility of the authentication systems. According to PTES, it emphasizes four guiding principles during this phase:

stealth: to avoid early detection by security tools.

speed: to maximize efficiency and test system resilience under pressure

depth: to dig deeper into access levels once login is successful

breadth: to explore systems and services that are vulnerable to brute-force methods.

By ethically performing a brute-force attack within a controlled test environment, the tester can not only know if access is possible but also how far can they potentially go once inside. They can also identify weak credentials, poor password policies, or rate-limiting failures that could be exploited by real attackers. (Ridwan, 2024)

Damn Vulnerable Web Application (DVWA) is an open-source PHP/MySQL-based web application that is purposefully designed to be insecure, serving as a training ground for security professionals, ethical hackers, developers, and students. It allows penetration testers to practice exploiting vulnerabilities legally while testing tools like Burp Suite, SQL map, and Hydra (S3Curiosity, 2023). In this phase, the identified weaknesses were actively exploited using multiple tools. The Damn Vulnerable Web Application (DVWA) was used to simulate a vulnerable login interface. Burp Suite is a web application security testing tool used by cybersecurity professionals to identify and exploit vulnerabilities in web applications. It functions as an intercepting proxy, allowing users to capture, inspect, and modify HTTP and HTTPS traffic between their browser and the target server (Sahu, 2024). Burp Suite and a proxy tool were used together to intercept, analyze, and modify the HTTP requests between the browser and the web application. These tools made it possible to capture hidden parameters like CSRF tokens and automate the login attempts using custom payloads. This setup successfully carried out a brute force attack by testing multiple credential combinations until the correct login was achieved.

vi. Post Exploitation

After the successful access during the exploitation phase, post-exploitation stage focuses on evaluating the impact of that access and understanding the security level of the compromised system where brute-force attacks were used to gain unauthorized access—this stage examines what could happen if an attacker maintained control over the breached accounts or systems.

According to the PTES framework, the main goal during this phase include identifying the business value and function of the compromised systems, checking if the attacker can create backdoors for future access, and exploring ways to remain undetected while maintaining control. These actions copy advanced attacker behavior, especially in internal penetration tests where the attacker exploits in long term. However, these actions must always stay within the limits agreed with the client. If additional weaknesses are discovered beyond the initial plan, the testing team must communicate transparently to avoid ethical or legal conflicts. This stage helps transitions into the most important aspect of any ethical hacking engagement—clear and professional reporting. (Ridwan, 2024)

After getting access, the Post-Exploitation phase aims to demonstrate the potential impact of a successful brute force attack. Tools like (msfconsole)and the Metasploit Framework were used to simulate a file upload vulnerability, allowing further interaction with the compromised system. These tools enabled the upload of reverse shells or scripts that could give deeper access to the target, proving how a simple brute force entry point could lead to full system compromise

vii. Reporting

Reporting is the final and one of the most important phase of a penetration test, where all findings are documented and shared to the organization. According to the PTES framework, a standard report consists of two key components. The executive summary is written for management and explains the goals, key findings, business impact, and overall security risks, along with recommendations. The technical report targets IT staff and security teams, offering insights into the vulnerabilities discovered, methods of exploitation, attack paths, and suggested recommendations. This phase ensures that the client understands both the risks and the steps needed to strengthen their overall security. (Ridwan, 2024)

3. Demonstration

In this section, demonstration of the practical execution of a brute force attack to showcase how attackers may exploit weak password security has been shown. The demonstration aims to provide a complete understanding of the steps involved in performing a brute force attack, the tools used, and the potential outcomes. It is important to emphasize that this demonstration is conducted within a customized and ethical environment, focusing strictly to legal guidelines and ethical norms. The purpose of this demonstration is to highlight the vulnerabilities associated with weak authentication mechanisms and to emphasize the importance of implementing robust security measures. The main sections defined by the standard as the basis for penetration testing execution:

a) Pre-engagement Interactions:
b) Intelligence Gathering:
c) Threat Modeling:
d) Vulnerability Analysis:
e) Exploitation:
f) Post Exploitation:
g) Reporting:

a) Pre-engagement interactions

TERMS OF REFRENCE

1. **Assignment Information**

Assignment title	Vulnerability Assessment and Penetration Testing (VAPT)
attack type	Brute force attack
duration	20 days
client	DVWA
Pen testers	LilaRaj Dura, Rebika Shrestha and Shreya Bastola

2. **Background**

2.1 Introduction

Between April 1- April 20, 2025, Radically Open DVWA, carried out a penetration test. This report contains our findings as well as detailed explanations of exactly how Brute force attack performed the penetration test.

2.2 Scope of work

The scope of the code audit was limited to the following targets:

- Clients (DVWA)
- Infrastructure components (monitoring & alert)
- Testing/profiling tools

The scoped services are broken down as follows:

- Attack: 8-10 days
- Reporting: 2-6 days
- (Optional) retest: 2-5 days
- Review: 2 days

- Total effort: 21 - 31 days

2.3 Project objectives

The main objective was to perform a security assessment of the Damn Vulnerable Web Application (DVWA), focusing specifically on the feasibility and impact of brute force attacks. The goal is to simulate real-world brute force scenarios targeting DVWA's login mechanisms and other authentication points. Through these tests, This aims to identify weaknesses in DVWA's handling of repeated login attempts, evaluate its resistance to automated attacks, and determine whether unauthorized access or privilege escalation is possible through brute force techniques.

Executive Summary

3. Methodology

3.1 Planning

Our general approach during penetration tests is as follows:

1. Reconnaissance

We attempt to gather as much information as possible about the target. Reconnaissance can take two forms:

active and passive. A passive attack is always the best starting point as this would normally defeat intrusion detection systems and other forms of protection afforded to the app or network. This usually involves trying to discover publicly available information by visiting websites, newsgroups, etc. An active form would be more intrusive, could possibly show up in audit logs and might take the form of a social engineering type of attack.

2. Enumeration

We use various fingerprinting tools to determine what hosts are visible on the target network and, more importantly, try to ascertain what services and operating systems they are running. Visible services are researched further to tailor subsequent tests to match.

3. Scanning

Vulnerability scanners are used to scan all discovered hosts for known vulnerabilities or weaknesses. The results are analyzed to determine if there are any vulnerabilities that could be exploited to gain access or enhance privileges to target hosts.

4. Obtaining Access

We use the results of the scans to assist in attempting to obtain access to target systems and services, or to escalate privileges where access has been obtained (either legitimately though provided credentials, or via vulnerabilities). This may be done surreptitiously (for example to try to evade intrusion detection systems or rate limits) or by more aggressive brute-force methods. This step also consists of manually testing the application. The discovered vulnerabilities from scanning and manual testing are moreover used to further elevate access on the application.

2.2 Risk Classification

Throughout the report, vulnerabilities or risks are labeled and categorized according to the Penetration Testing Execution Standard (PTES).

These categories are:

- Moderate:
  Moderate risk of security controls being compromised with the potential for limited financial/reputational losses occurring as a result.
- Low:
  Low risk of security controls being compromised with measurable negative impacts as a result.

b. Intelligence Gathering:

Gather the information about DVWA in web browser.

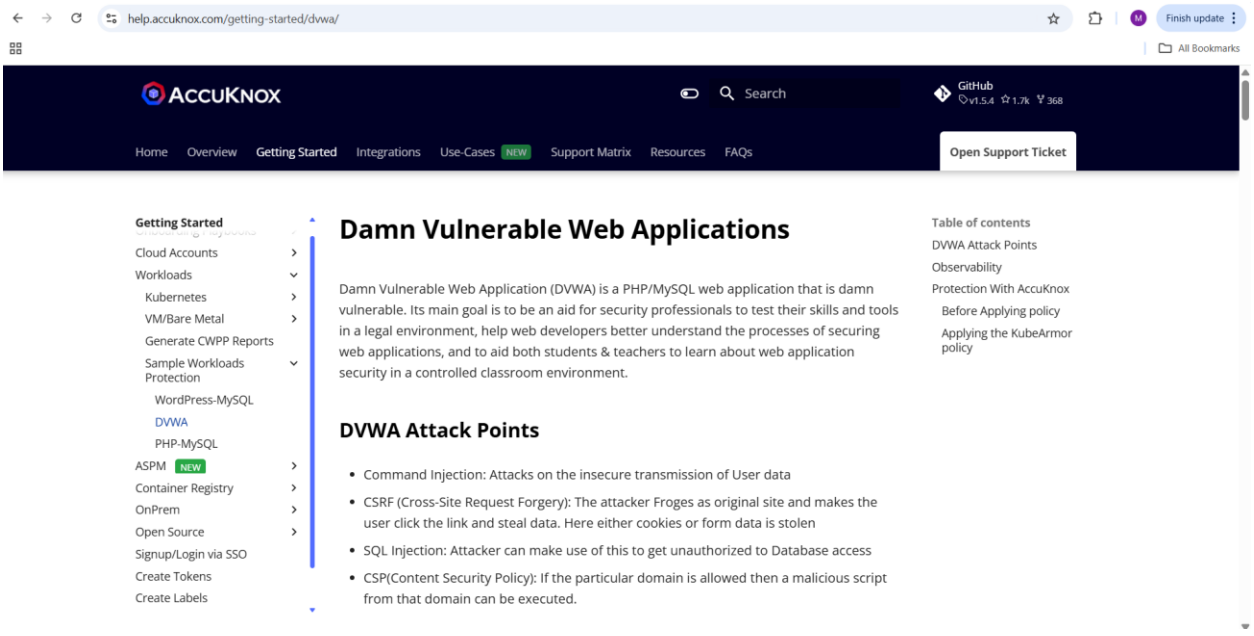

Figure 4:About DVWA

Gathering the IP address of metasploitable 2 using command ifconfig.

```
To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
No mail.
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:a1:cc:f2
          inet addr:10.0.2.4  Bcast:10.0.2.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fea1:ccf2/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:36 errors:0 dropped:0 overruns:0 frame:0
          TX packets:67 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:4709 (4.5 KB)  TX bytes:6926 (6.7 KB)
          Base address:0xd020 Memory:f0200000-f0220000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:95 errors:0 dropped:0 overruns:0 frame:0
          TX packets:95 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:19669 (19.2 KB)  TX bytes:19669 (19.2 KB)

msfadmin@metasploitable:~$ _
```

Figure 5:IP address of metasploitable 2.

Gathering the IP address of kali linux using command ifconfig.

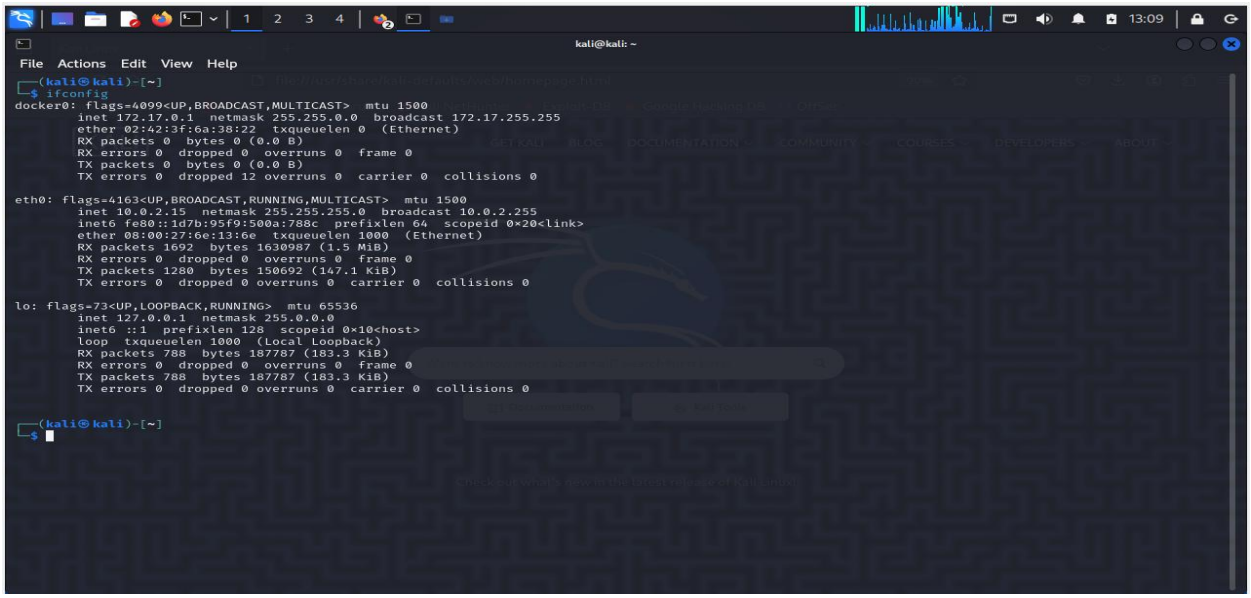

Figure 6:IP address of kali linux.

Using nmap tool, the information about port open or close. The port of ftp, ssh,http are open state.

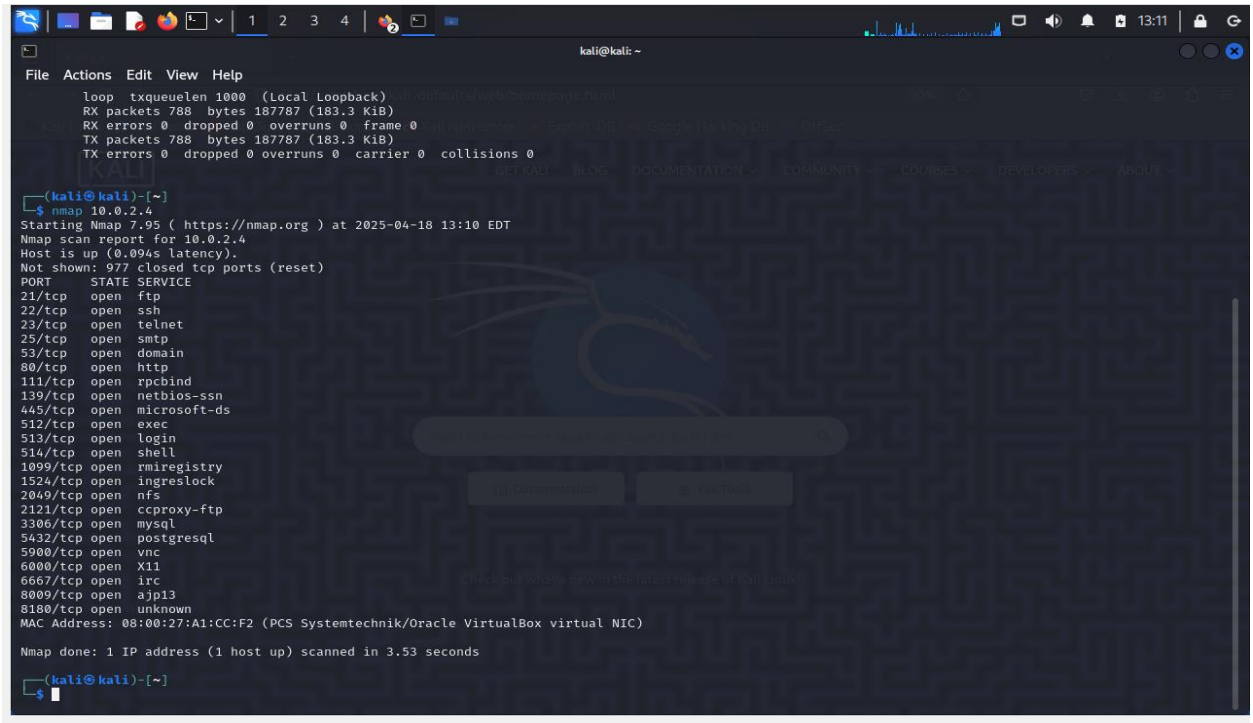

Figure 7:Open ports.

c. Threat Modeling:

Vulnerability Analysis:

Executing a scan to a metasploitable 2 (127.0.0.1).

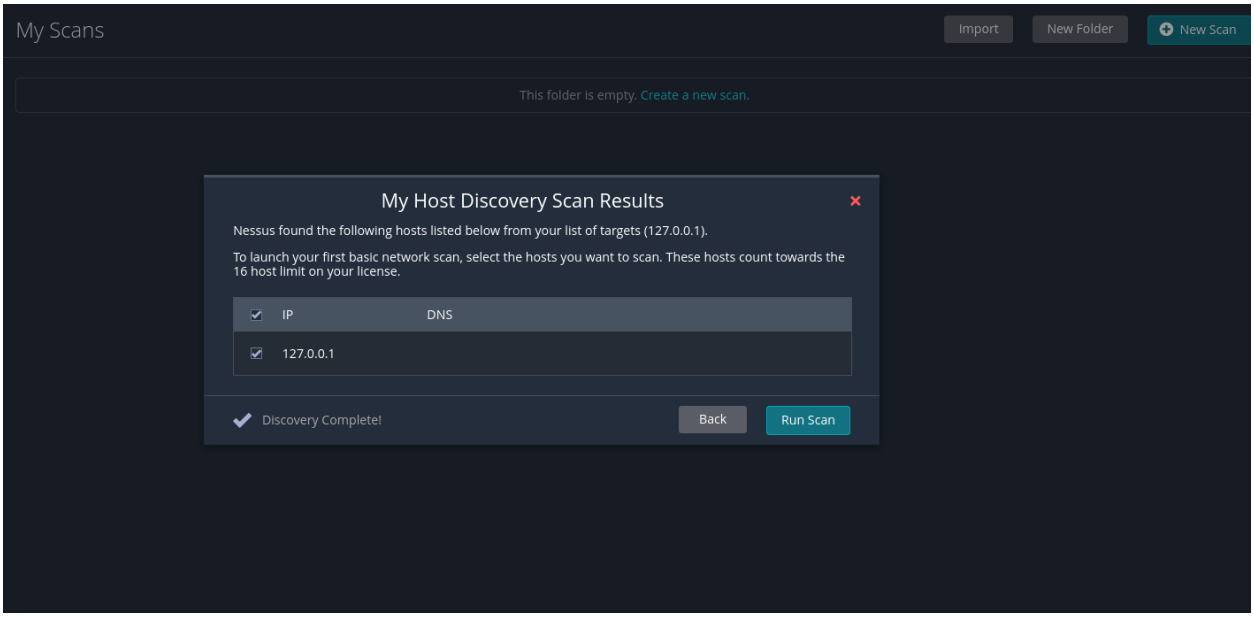

Figure 8: Scan of 127.0.0.1

Nessus found 59 vulnerabilities.

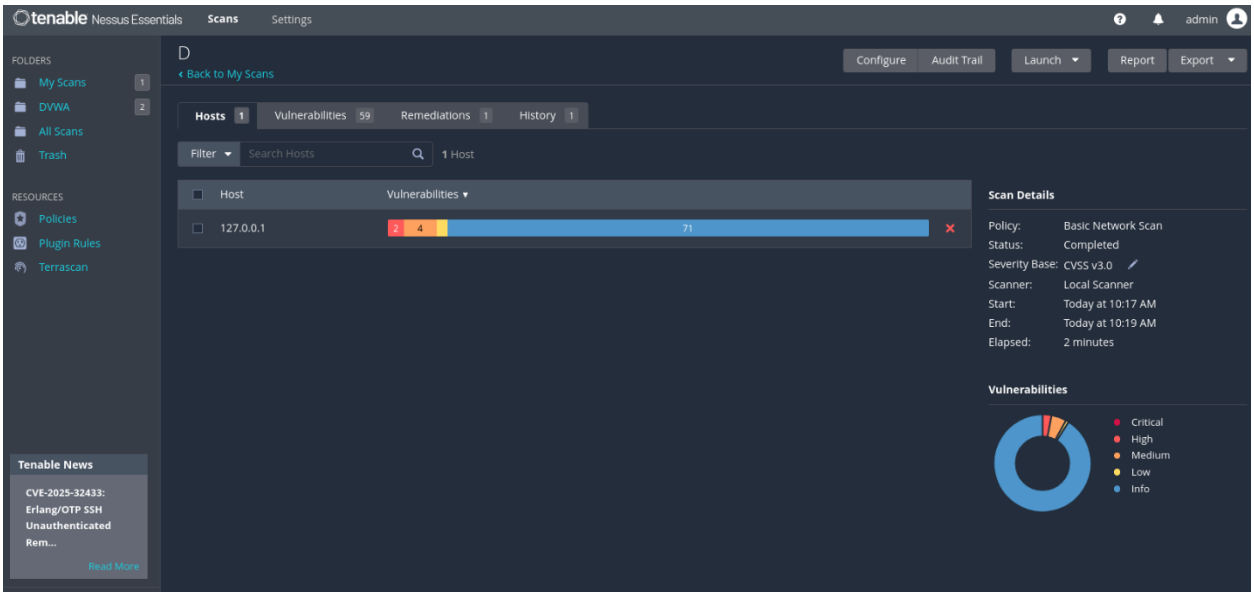

Figure 9: Overall Nessus scan result (Number of vulnerabilities)

Some are categorized as critical and high-risk vulnerabilities.

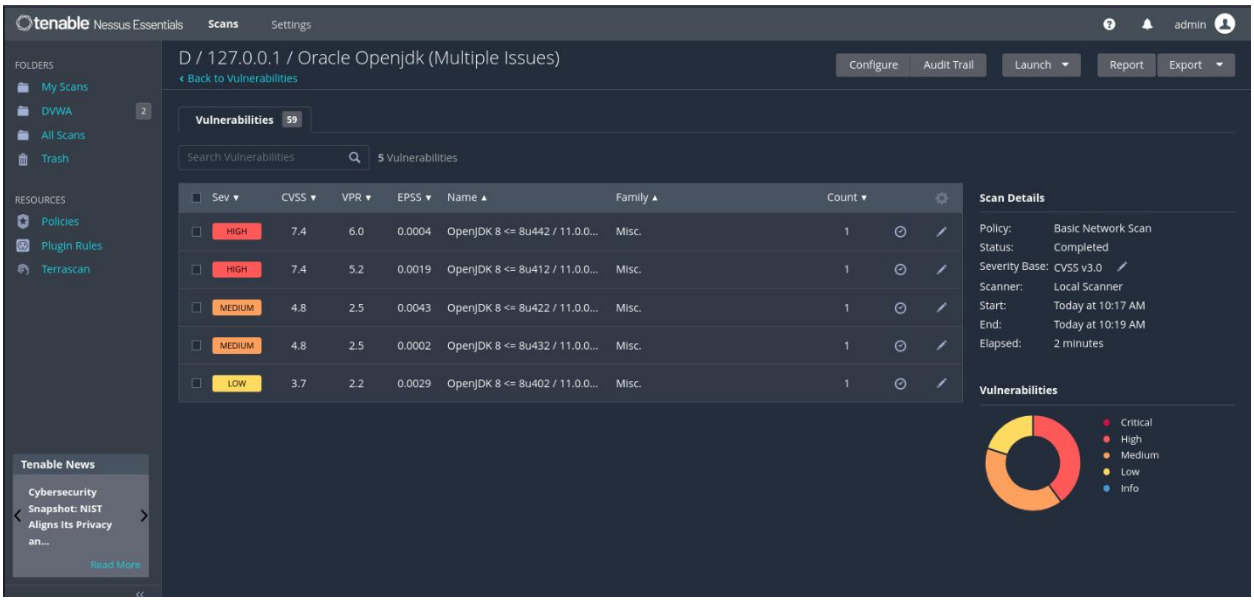

Figure 10: High and medium category vulnerabilities.

Most of them are categorized as INFO (informational), and some MIXED (which could contain medium/high severity items).

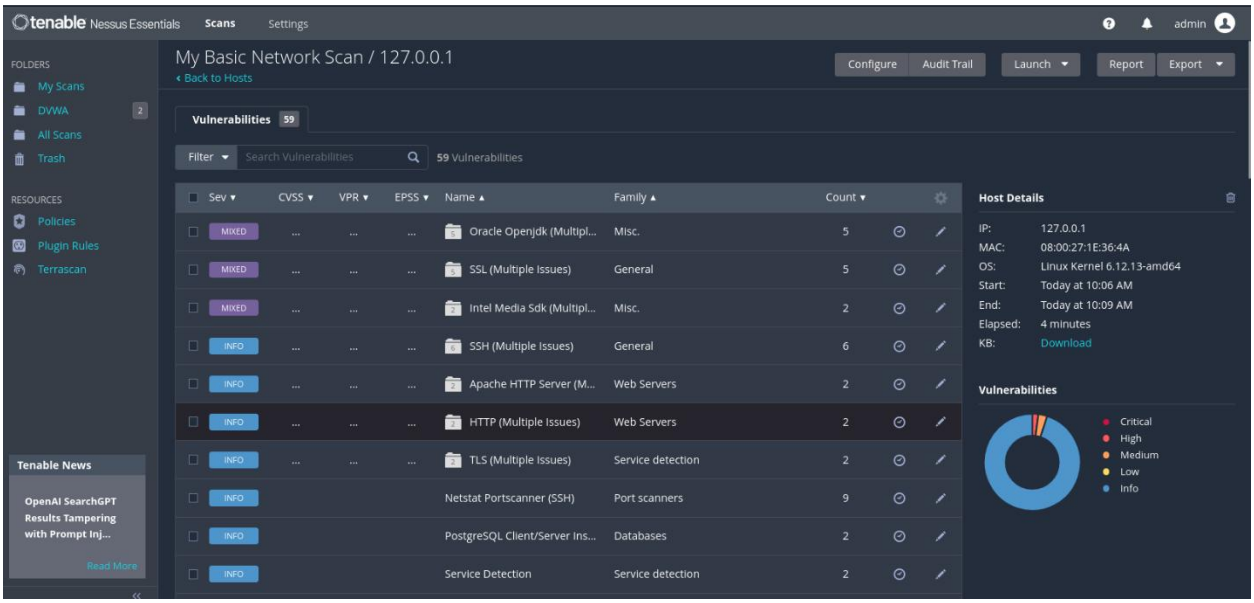

Figure 11: Mixed and informational vulnerabilities.

Gathering the information about Hyper Text transfer protocol (HTTP).

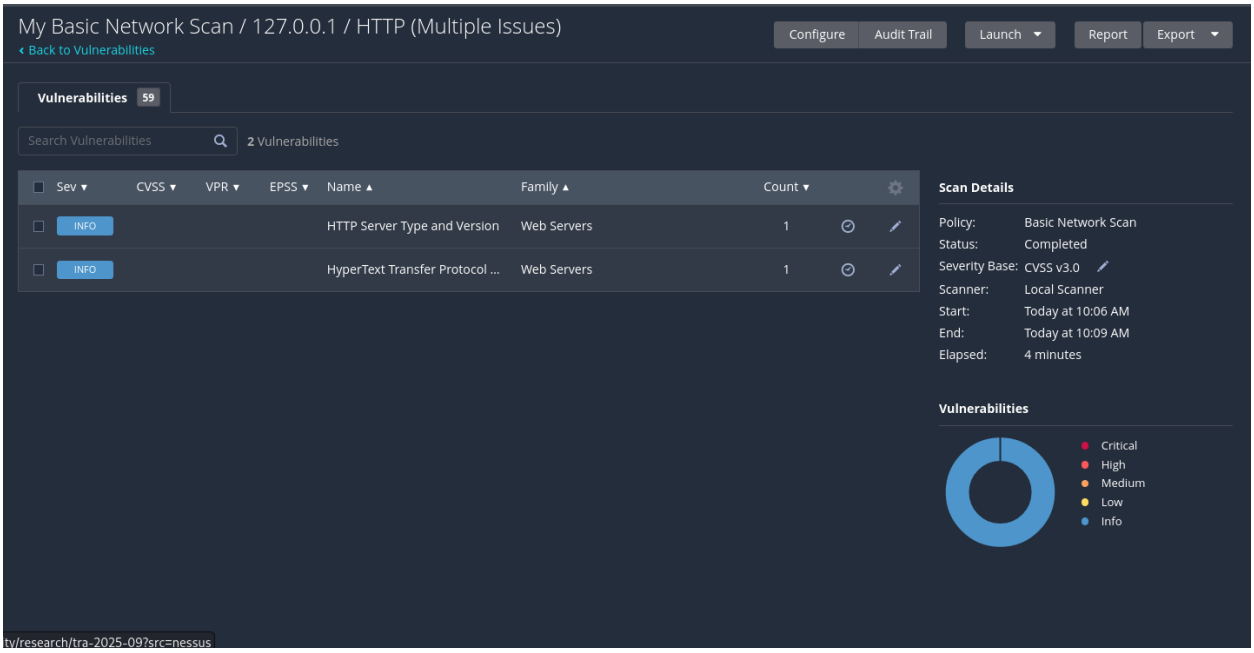

Figure 12: Vulnerabilities information of HTTP.

Information of HTTP.

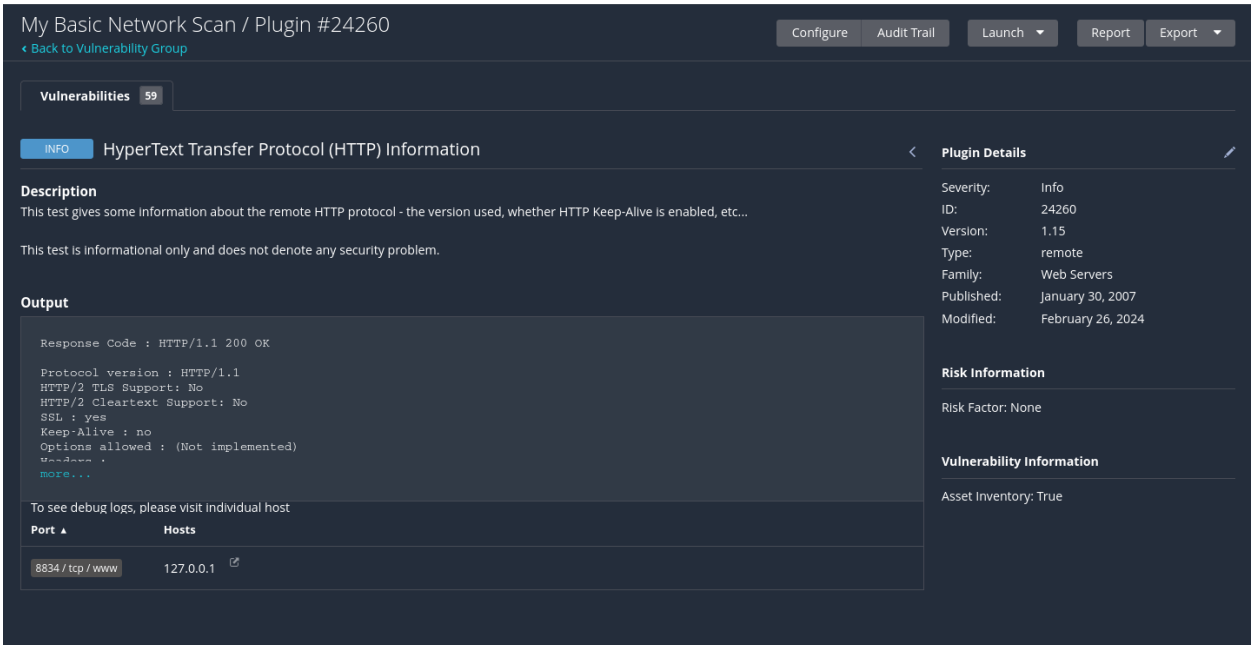

Figure 13: Information about HTTP.

Information about vulnerabilities of SSL.

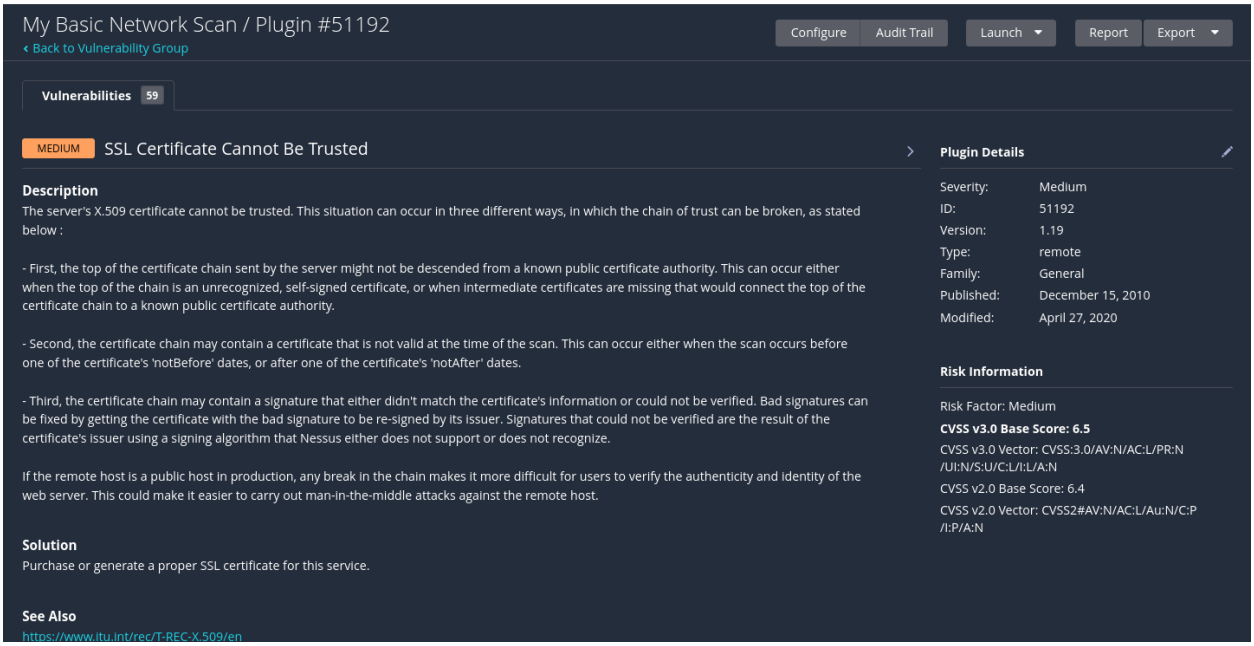

Figure 14: Information about SSL.

d. Exploitation:

Proxy extension is created. Hostname=127.0.0.1 and port=8080.

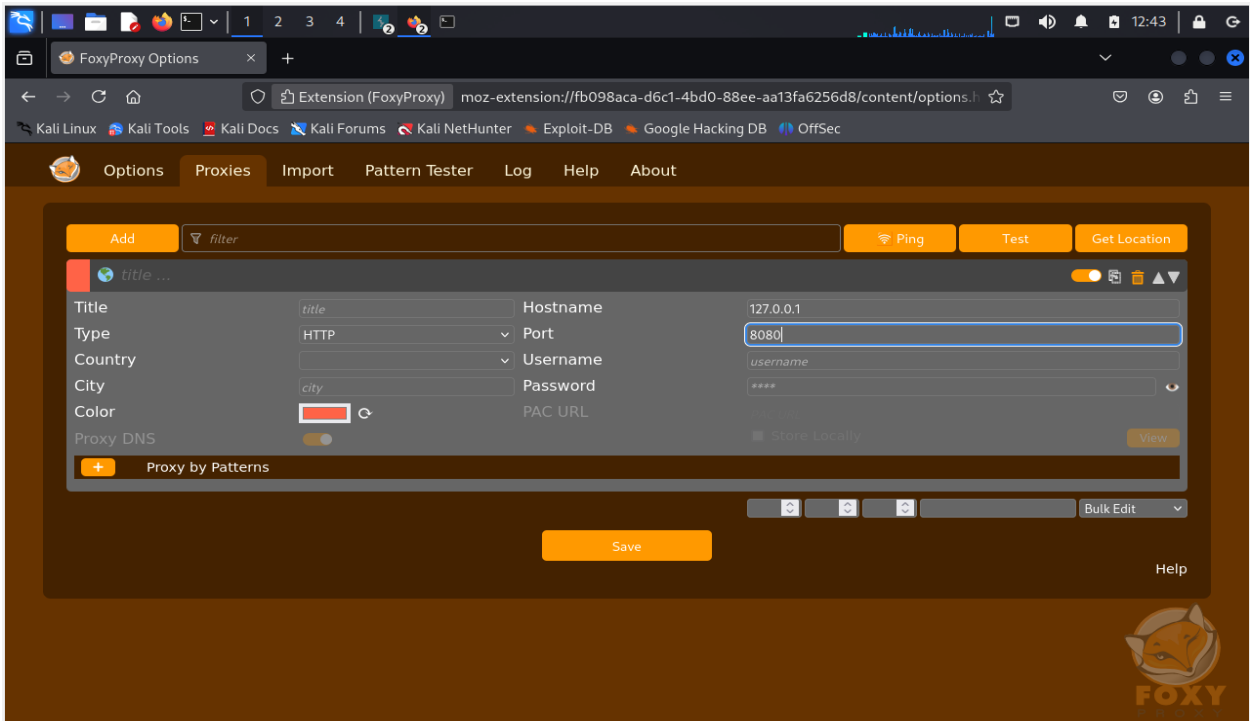

Figure 15: Proxy

Interception is turned on.

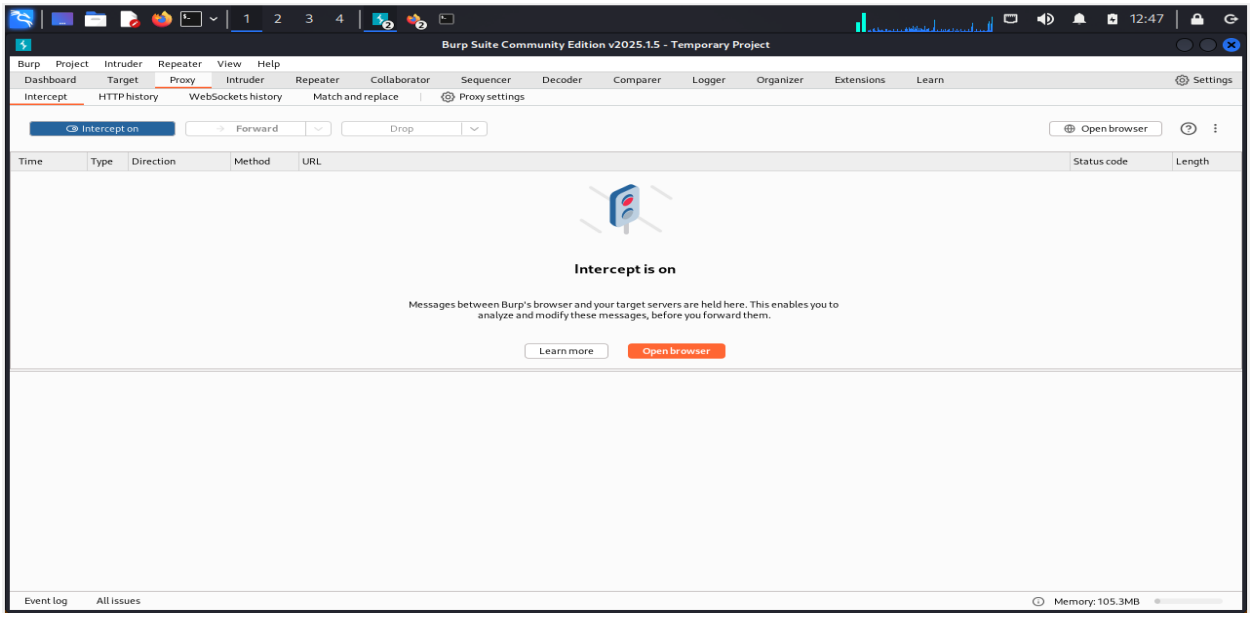

Figure 16: Interception.

Downloading CA certificate.

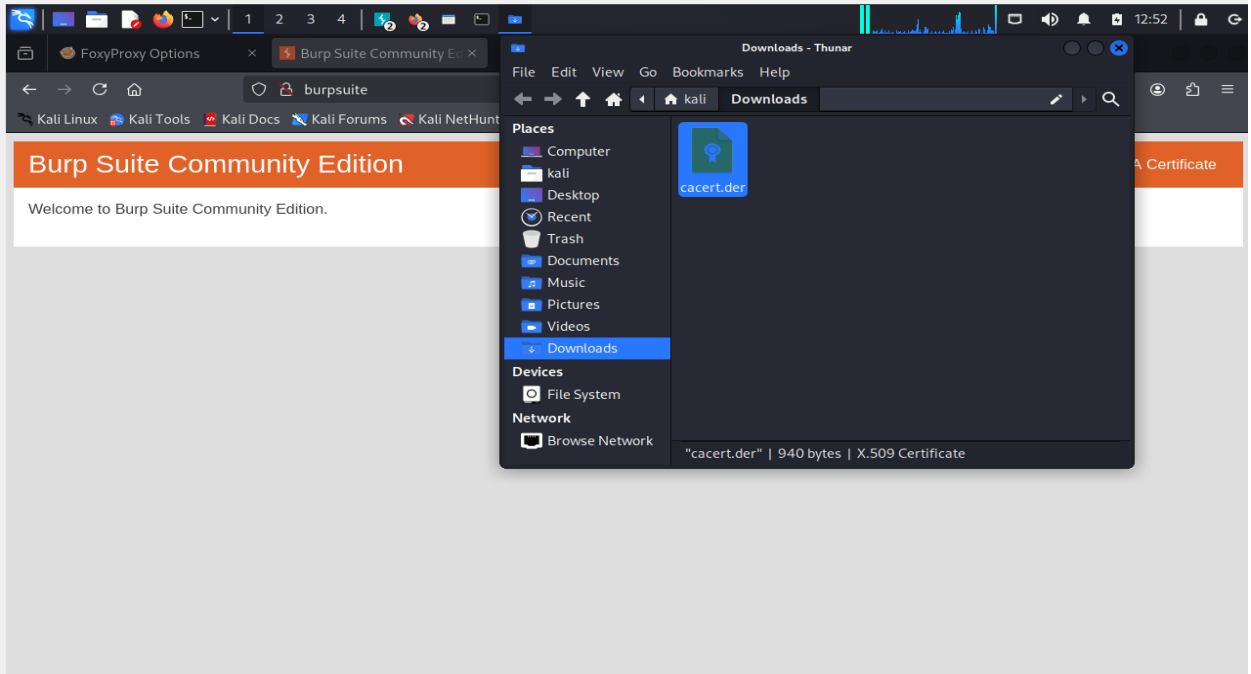

Figure 17: CA certificate downloading.

Importing ca certificate into web browser.

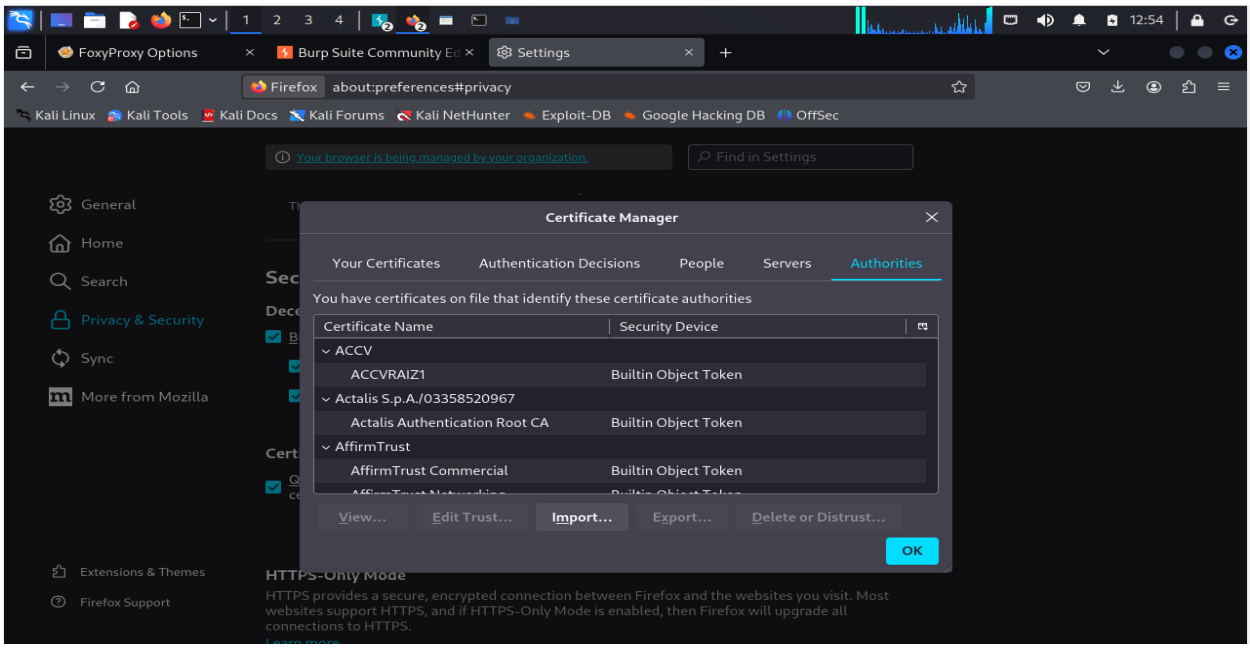

Figure 18:Certificate manager.

Select both check and click ok.

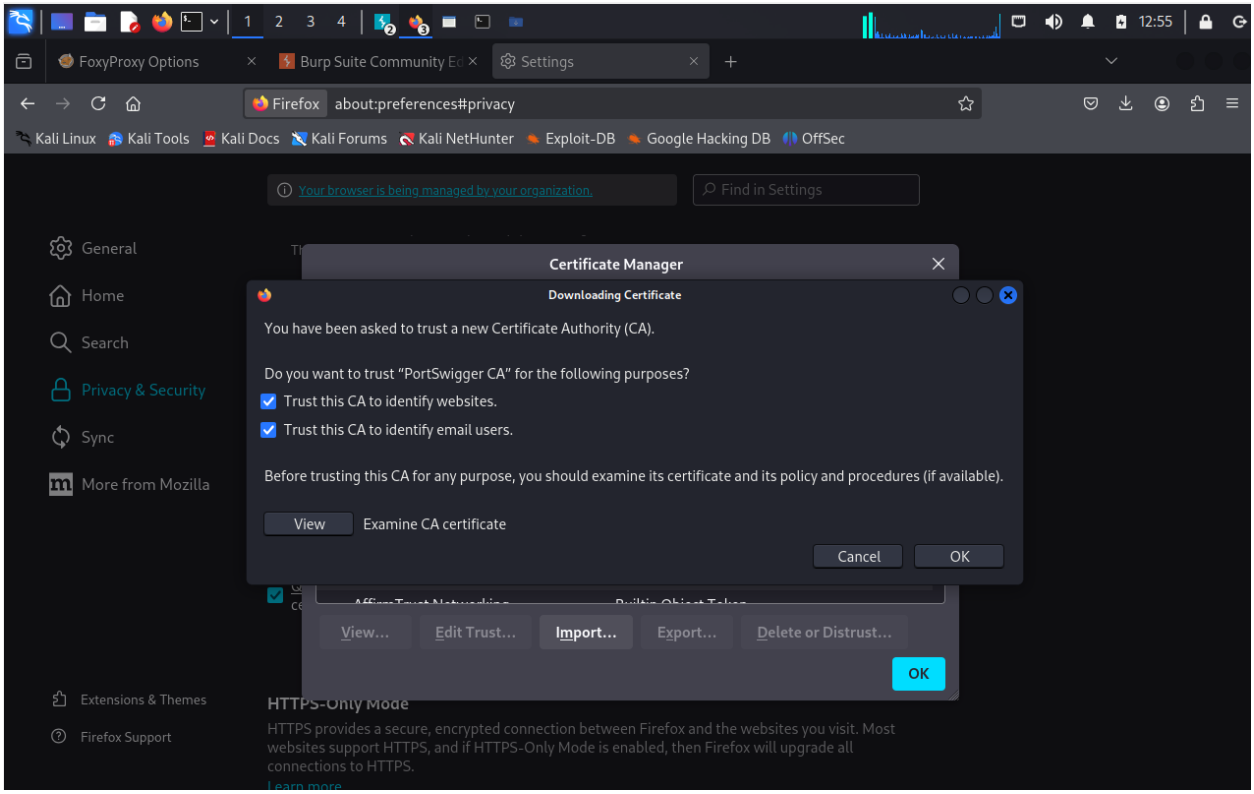

Figure 19: Downloading ca certificate into web browser.

Opening DVWA.

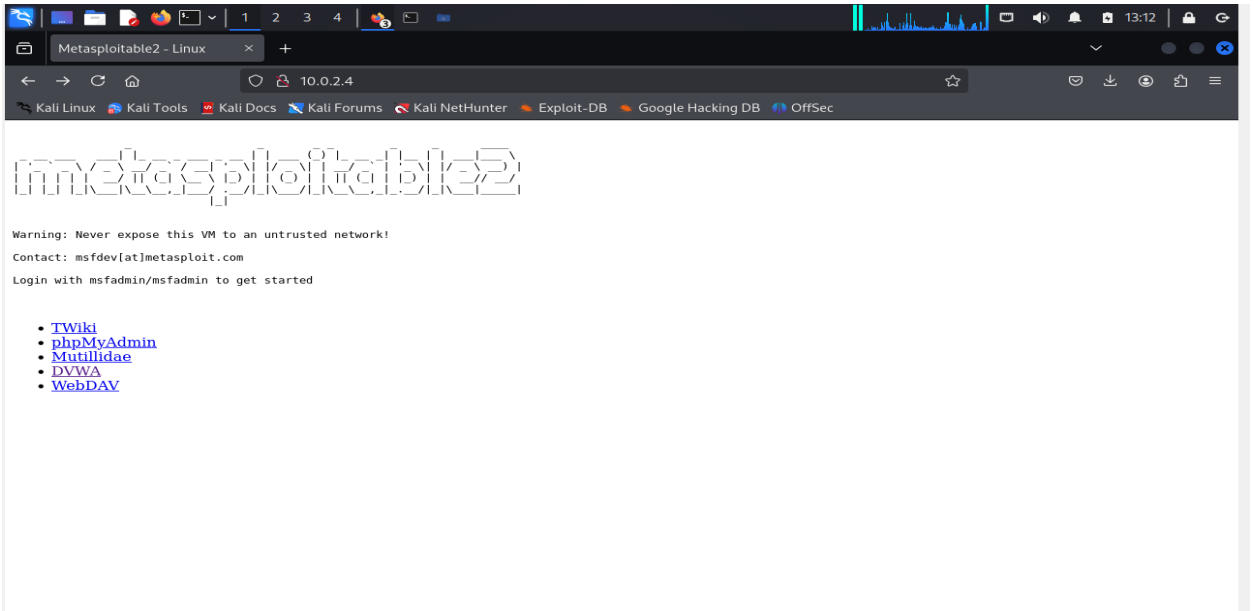

Figure 20: Metasploitable 2

Setting the security of the DVWA into low.

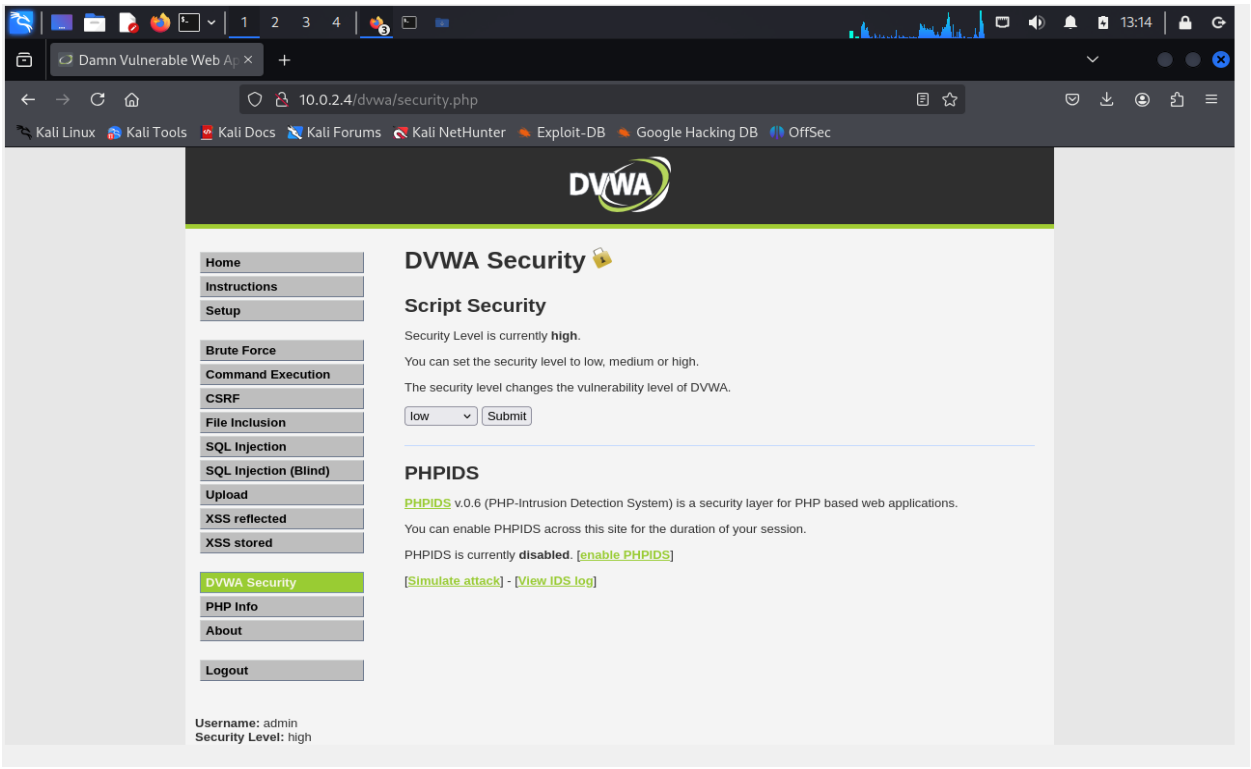

Figure 21:DVWA setting in low security.

Interception is on through brup suite.

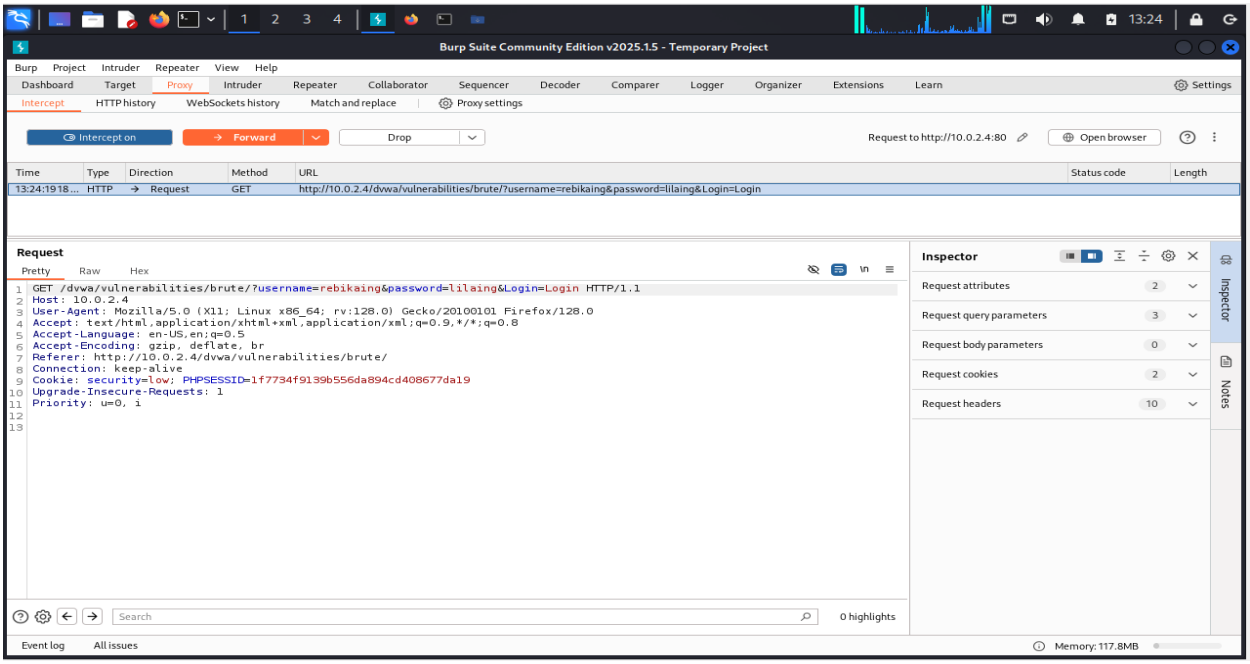

Figure 22:Request got intercepted.

Select username and password and click add button.

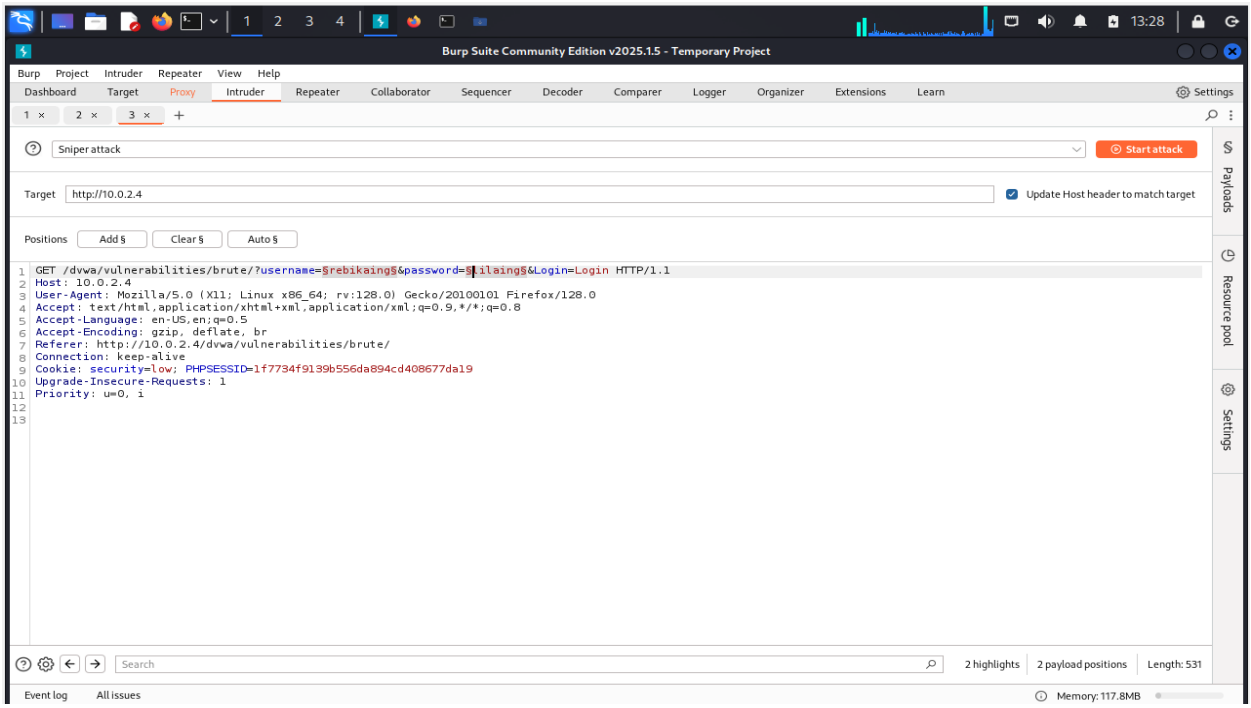

Figure 23: Adding marker into username and password.

Select the attack type into cluster bomb.

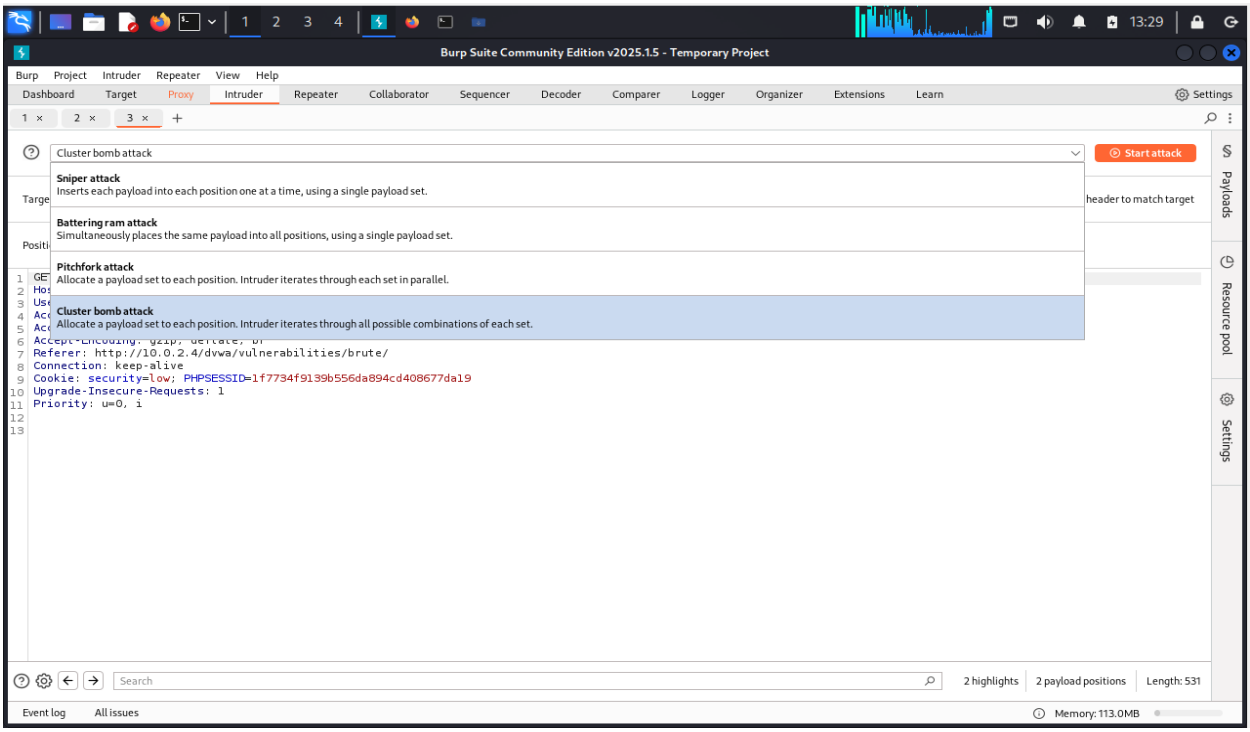

Figure 24: attack type into cluster bomb.

Click proxy and select payload position in 1 for username select payload type into runtime file.

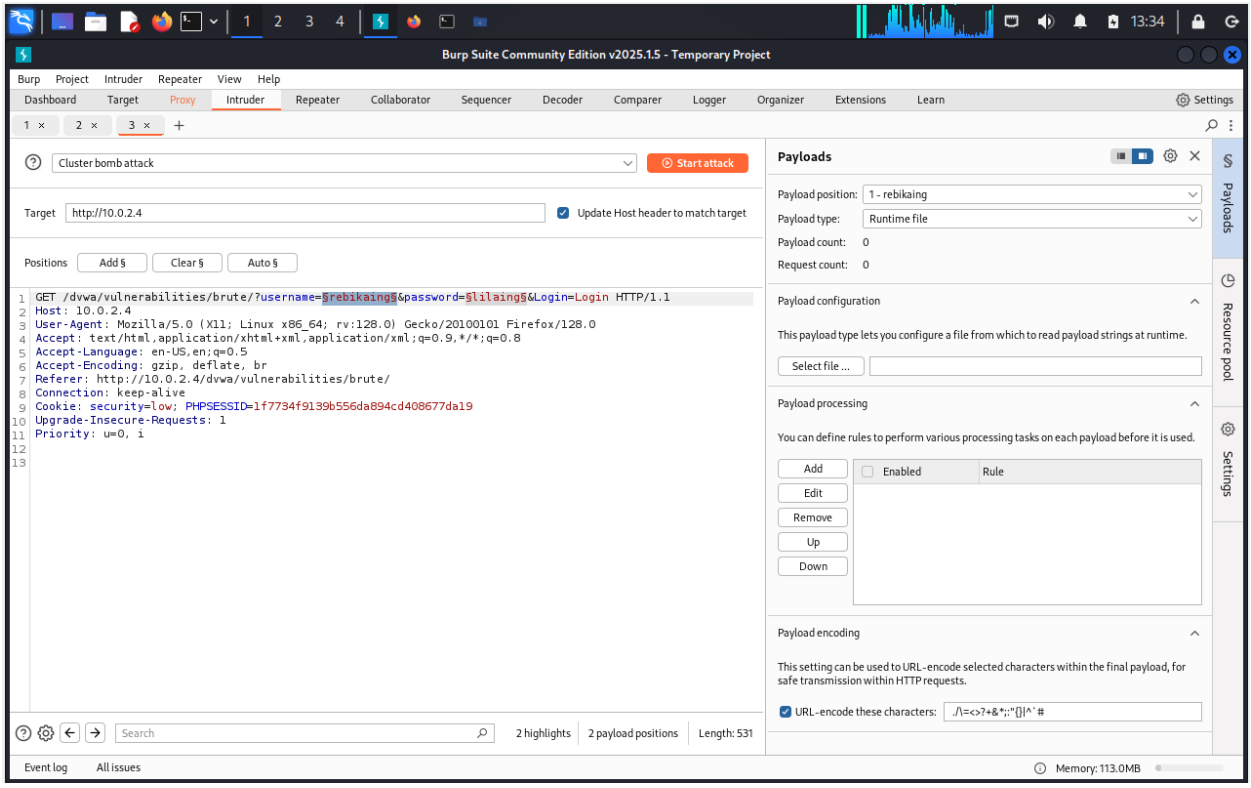

Figure 25:Payload for username.

Click on select file and direct into the wordlist file.

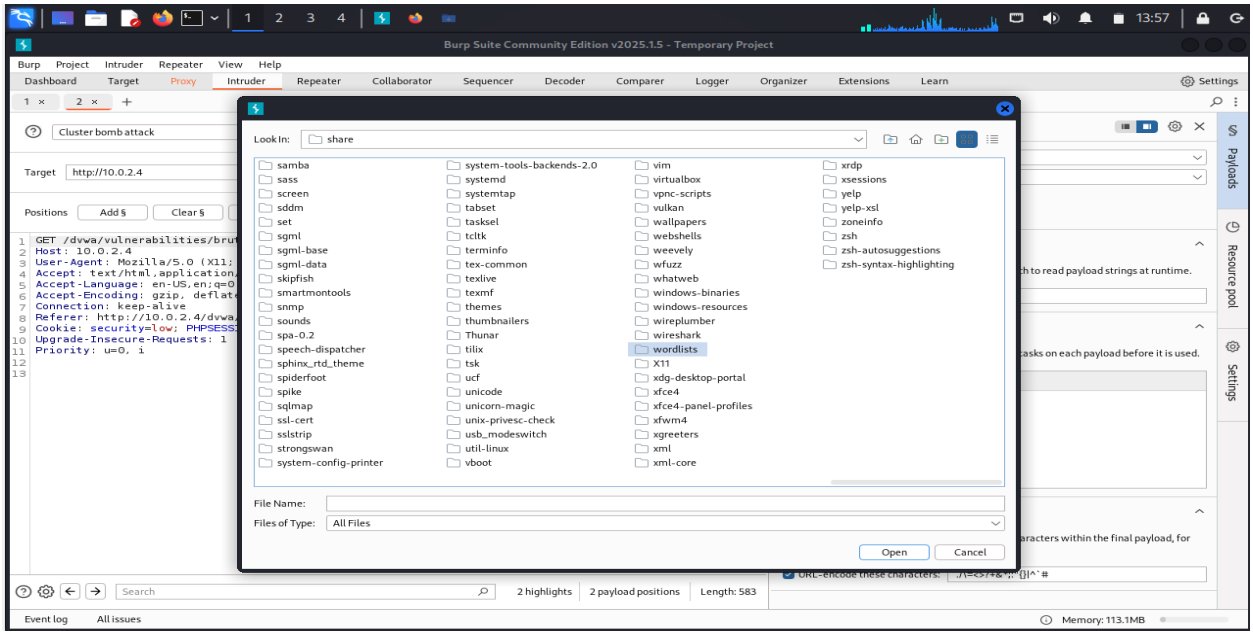

Figure 26:Directing into wordlist folder.

Select users.txt file for the dictionary file.

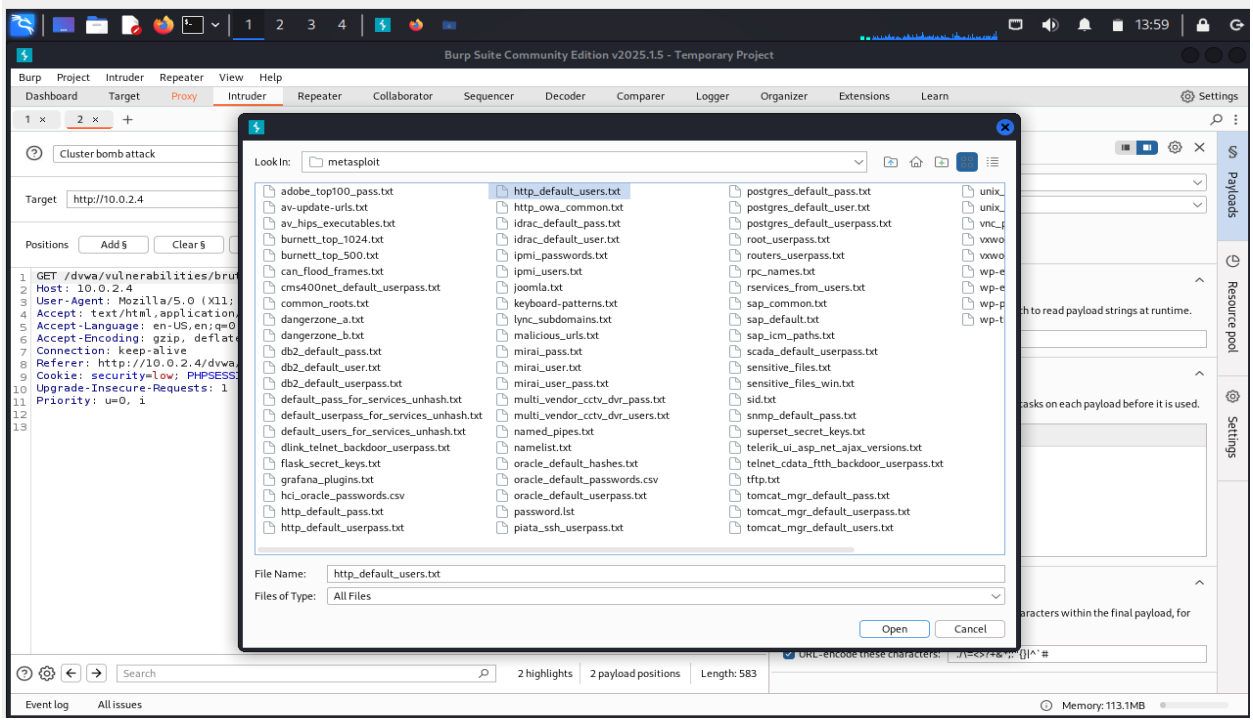

Figure 27: Dictionary file for username.

Do same for password filed, set payload position into 2 and type in runtime files.

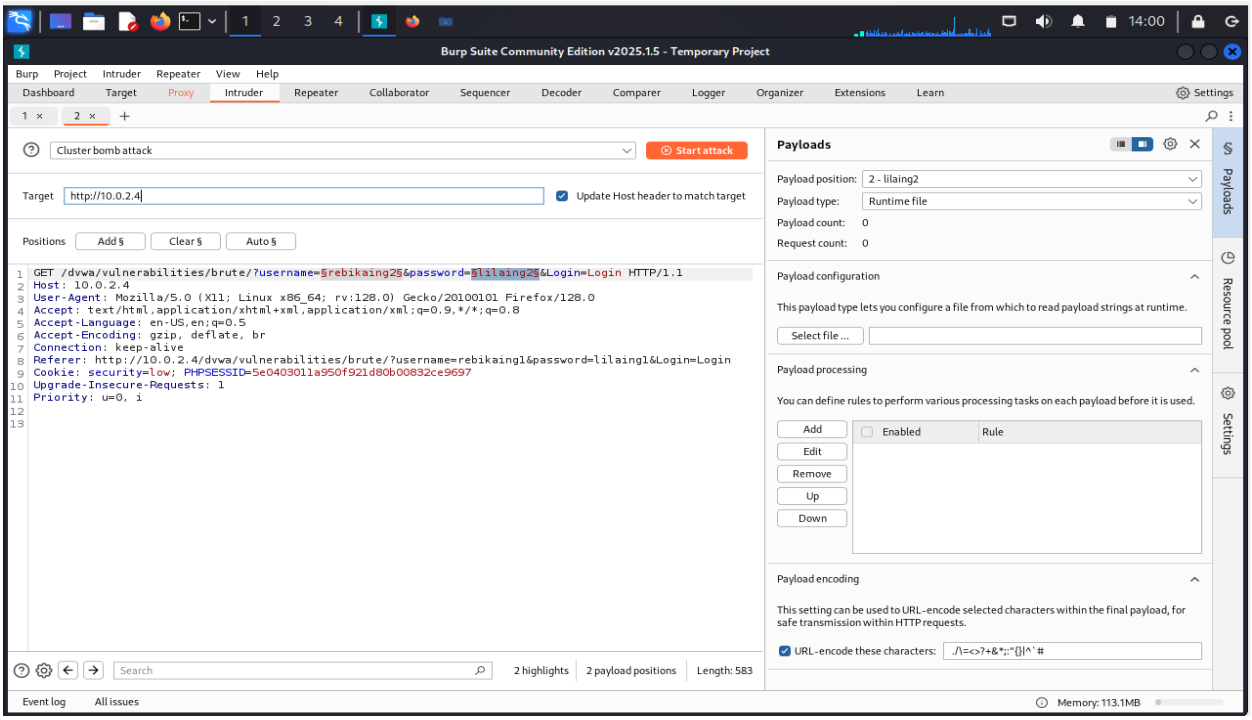

Figure 28: Setting payload for password field.

Direct into the dictionary file for password.

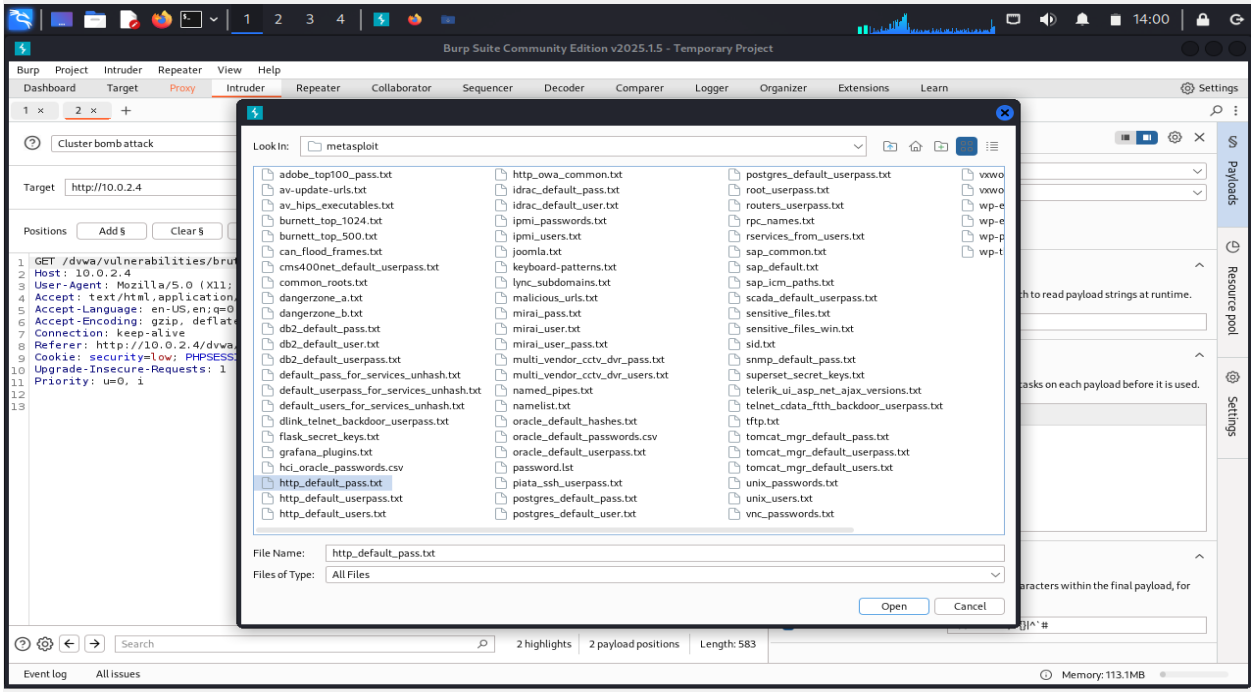

Figure 29: Dictionary file for password.

Select Run attack and check the result, we can conform the username and password by the content length where it will chagne the content length.

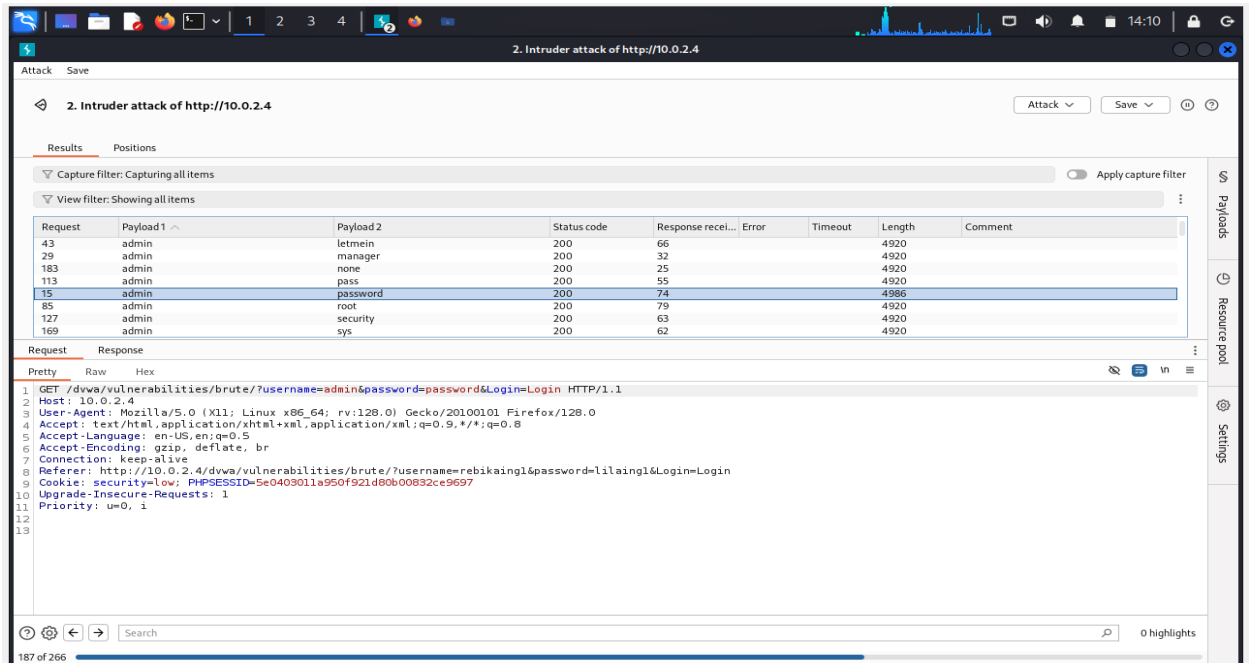

Figure 30: Guessing the username and password by content length.

Checking the username and password from the attack to login into website.

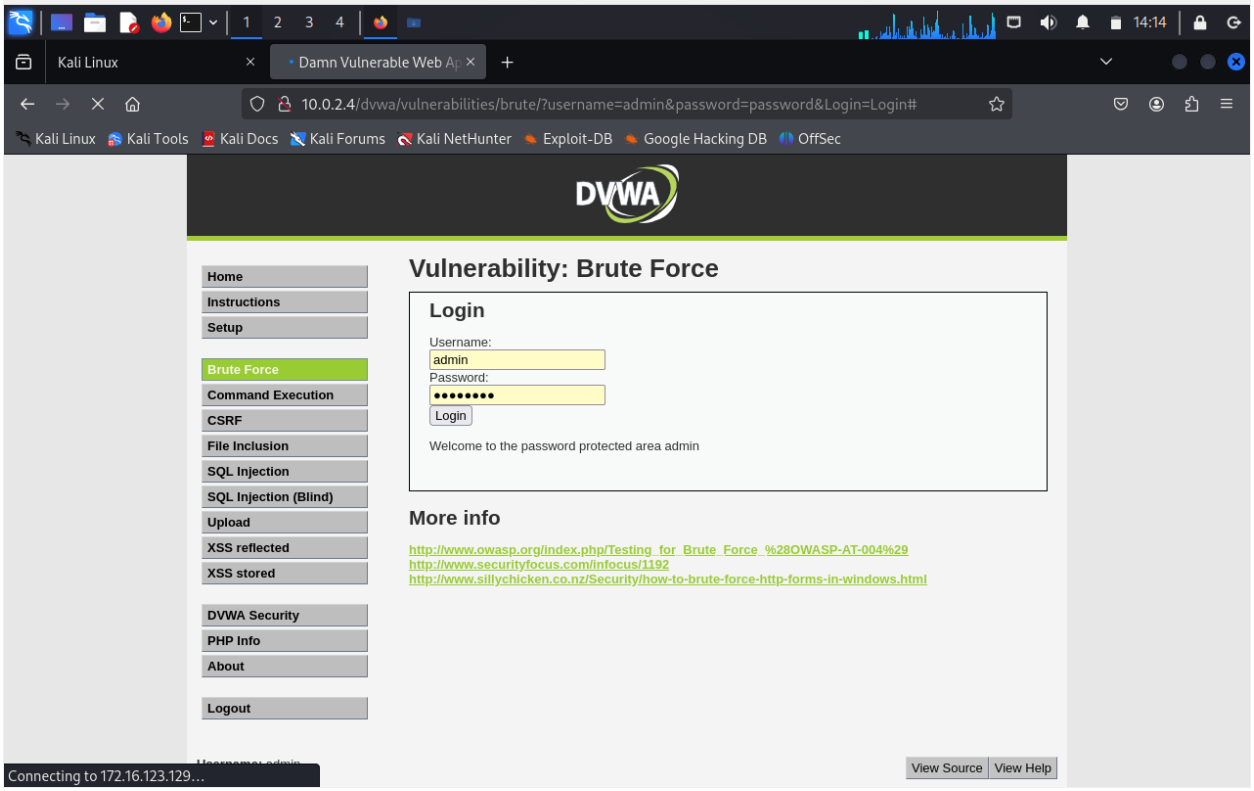

Figure 31:Trying username and password from attack to login.

Dictionary files can be created by using CRUNCH.

Figure 32:Using CRUNCH.

Customized dictionary can be created using crunch.

Figure 33:Customized Dictionary.

Setting DVWA security level into medium.

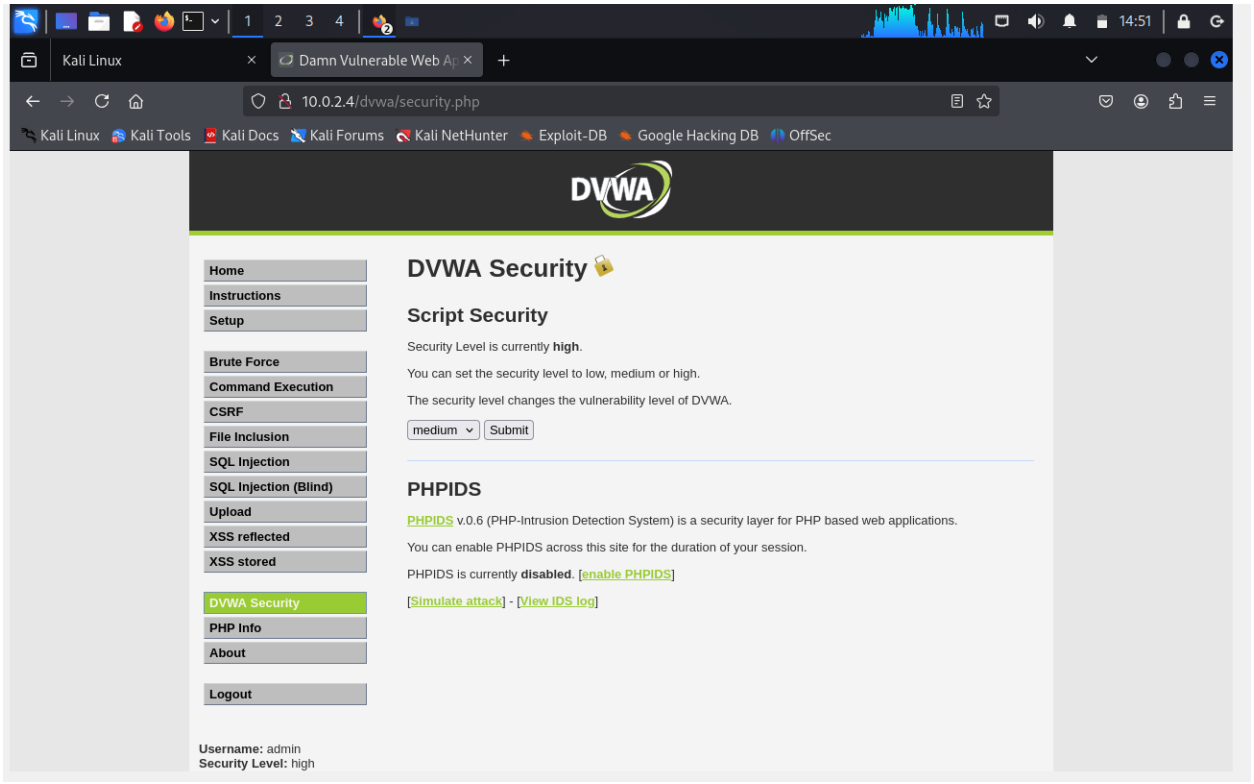

Figure 34:Security level medium.

Interception is on.

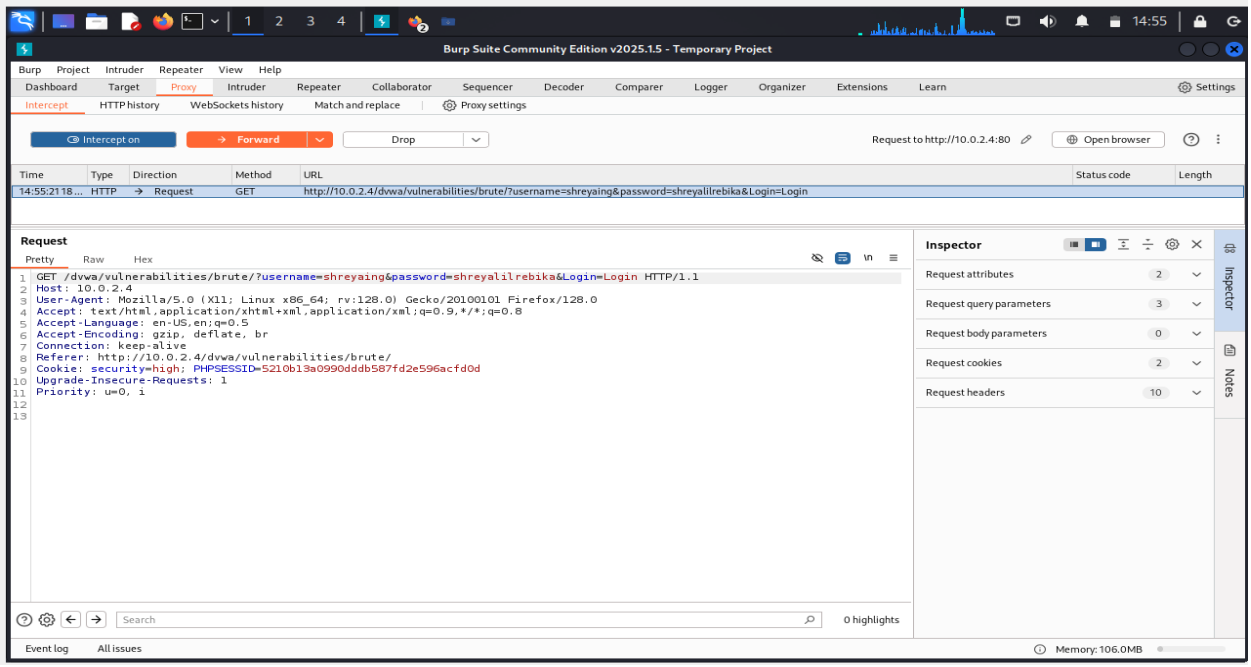

Figure 35:Interception is on and getting details.

Setting payload position into 2 and type into runtime file after adding hashtag into username and password.

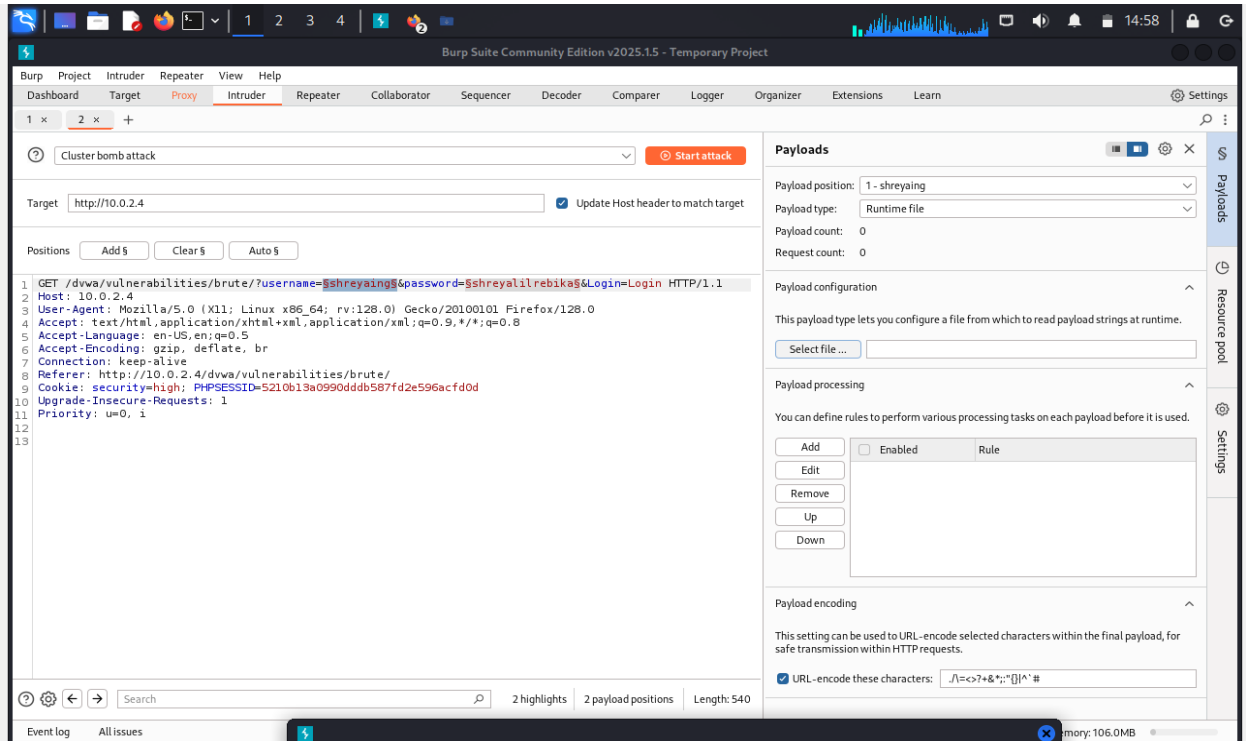

Figure 36:Payload 1.

Directing to the dictionary file.

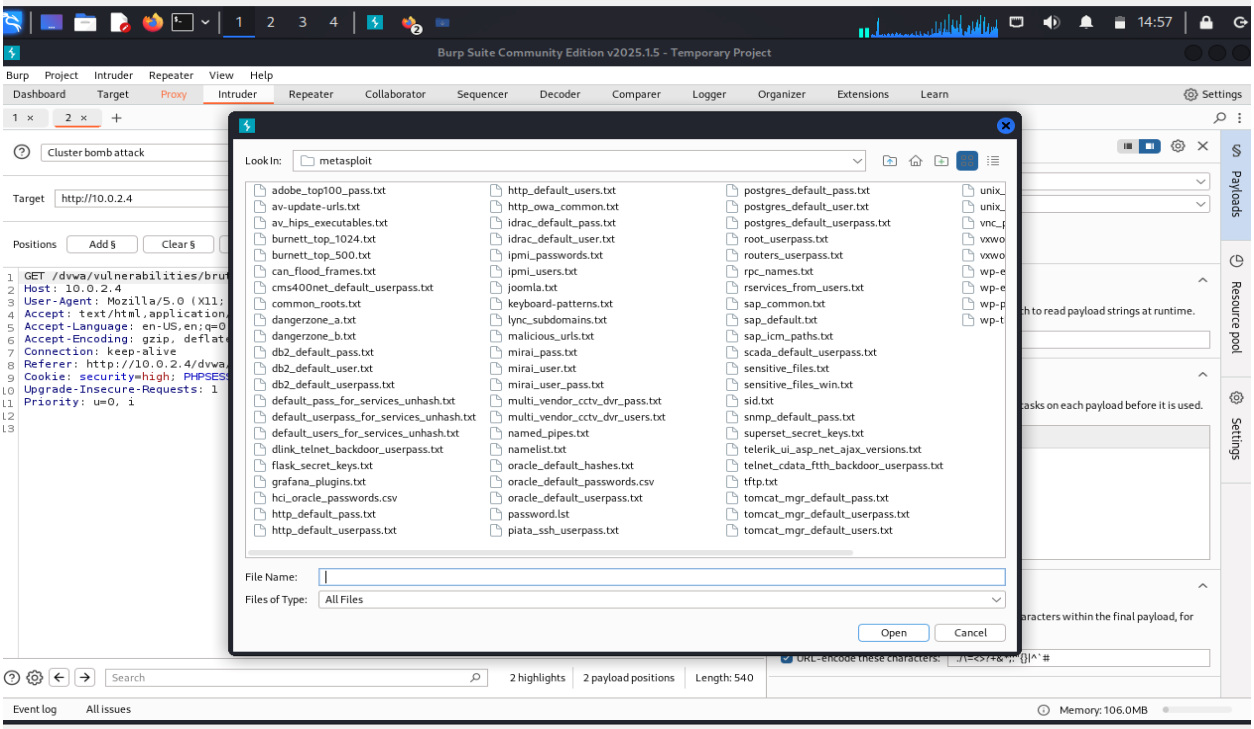

Figure 37:Dictionary file for payload 1.

Setting payload 2 for password.

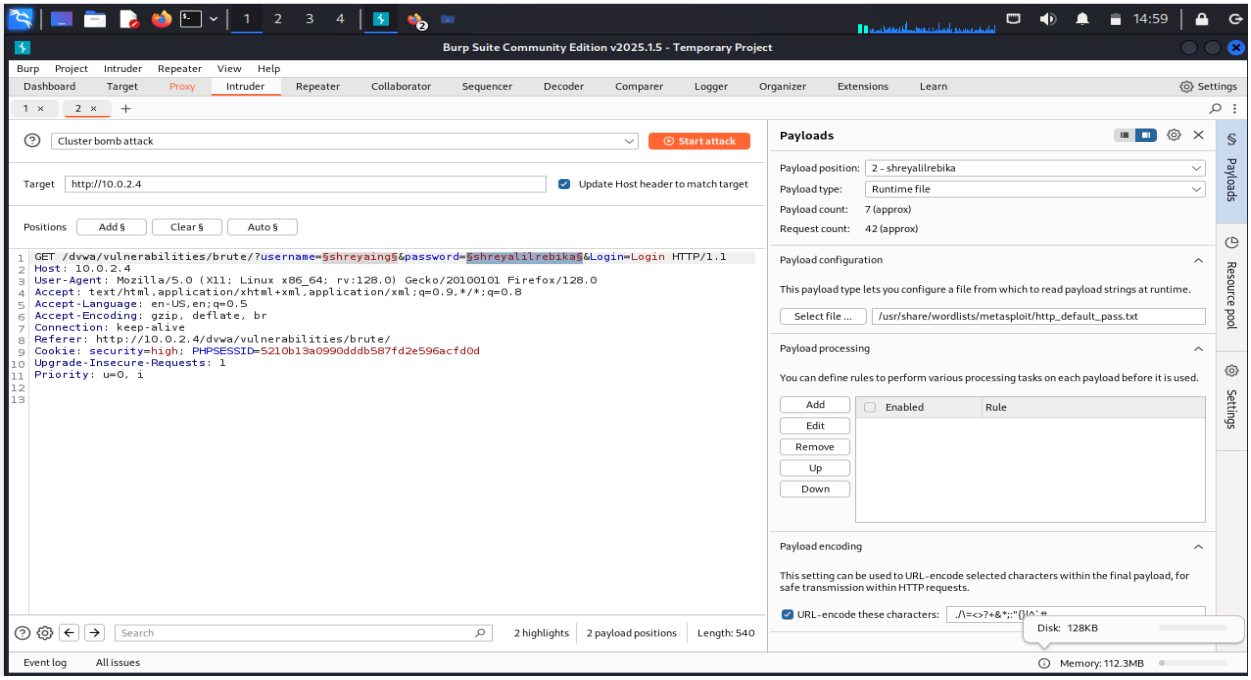

Figure 38:Setting for payload for password field.

Directing Dictionary file for password field.

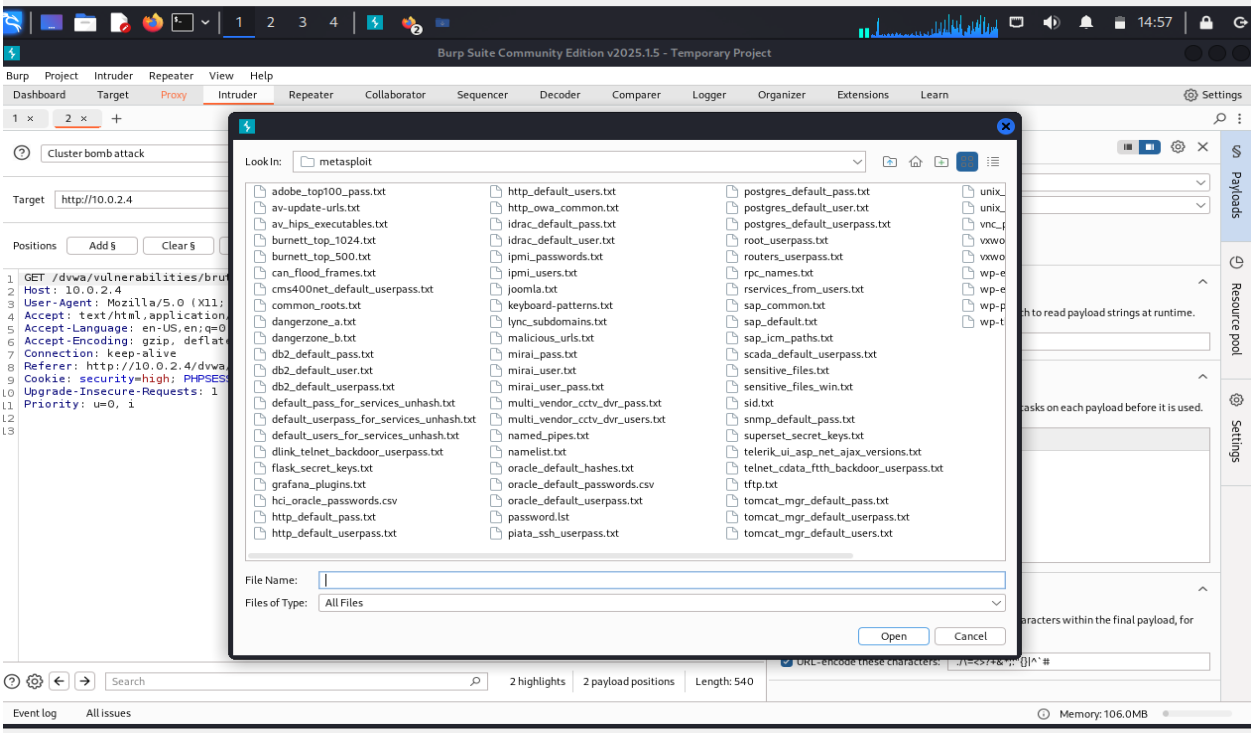

Figure 39: Directing Dicitionary file.

Start the Attack and monitor the length of credentials. When it fluctuates it might be the username and password for website.

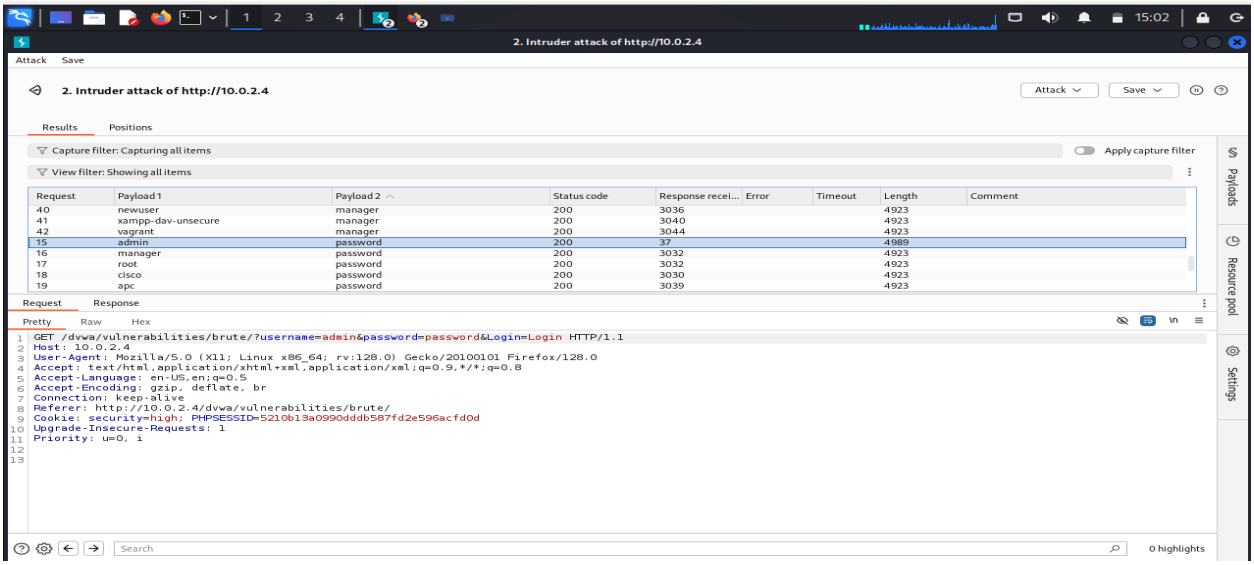

Figure 40: After the attack

e. Post Exploitation:

After knowing the credential like username and password for login. The post exploitation is done by uploading files into the website.

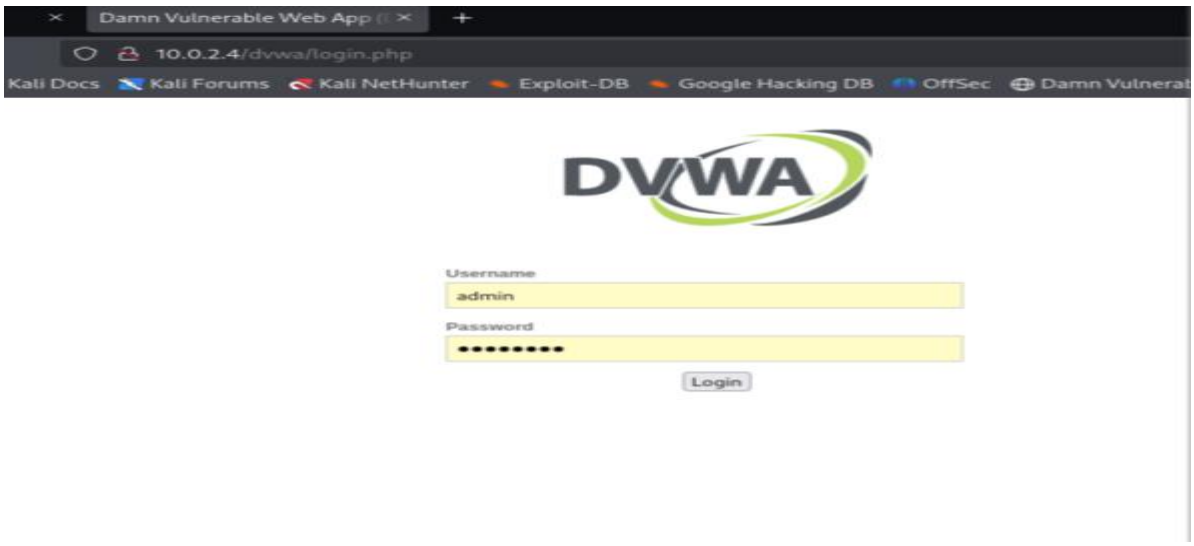

Figure 41: Login using credentials from attack.

Setting security into low.

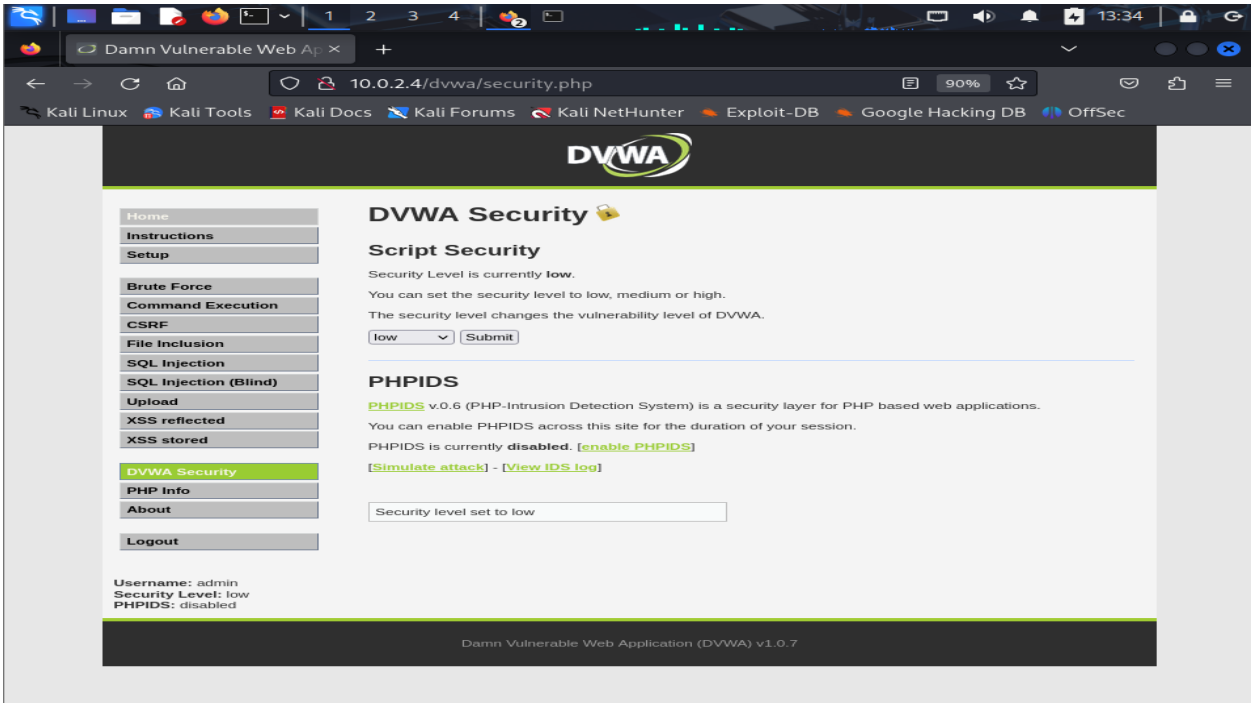

Figure 42: Setting security into low.

Uploading file.

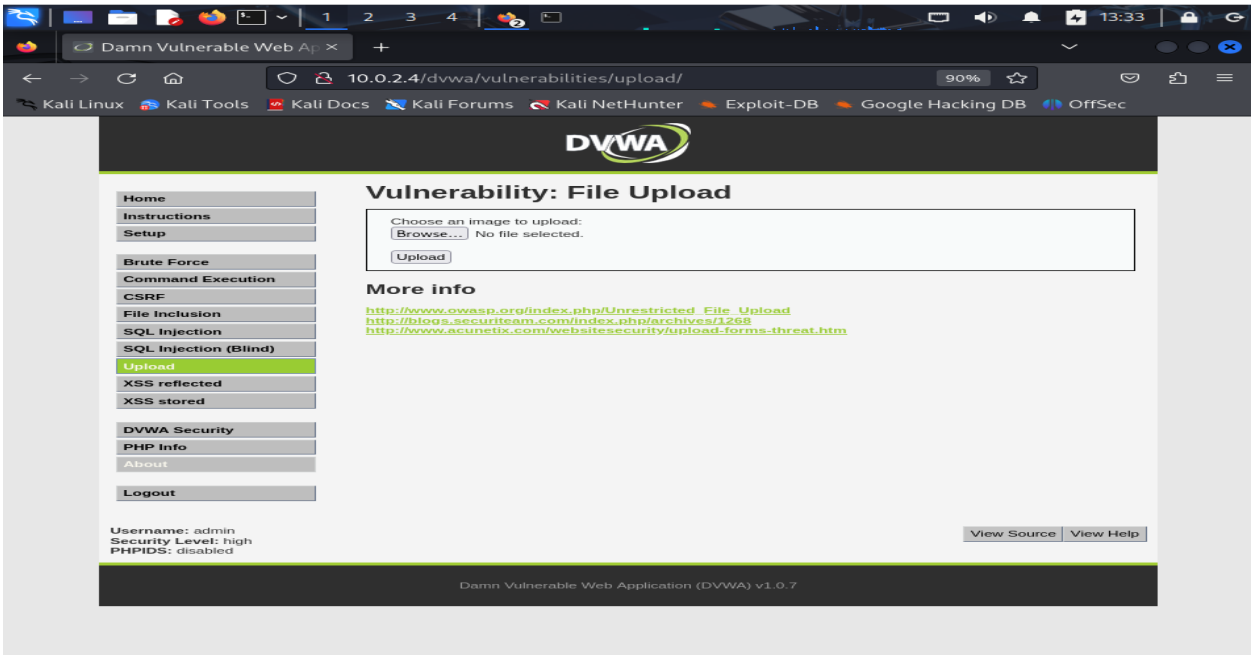

Figure 43:Uploading file.

Navigating to the file.

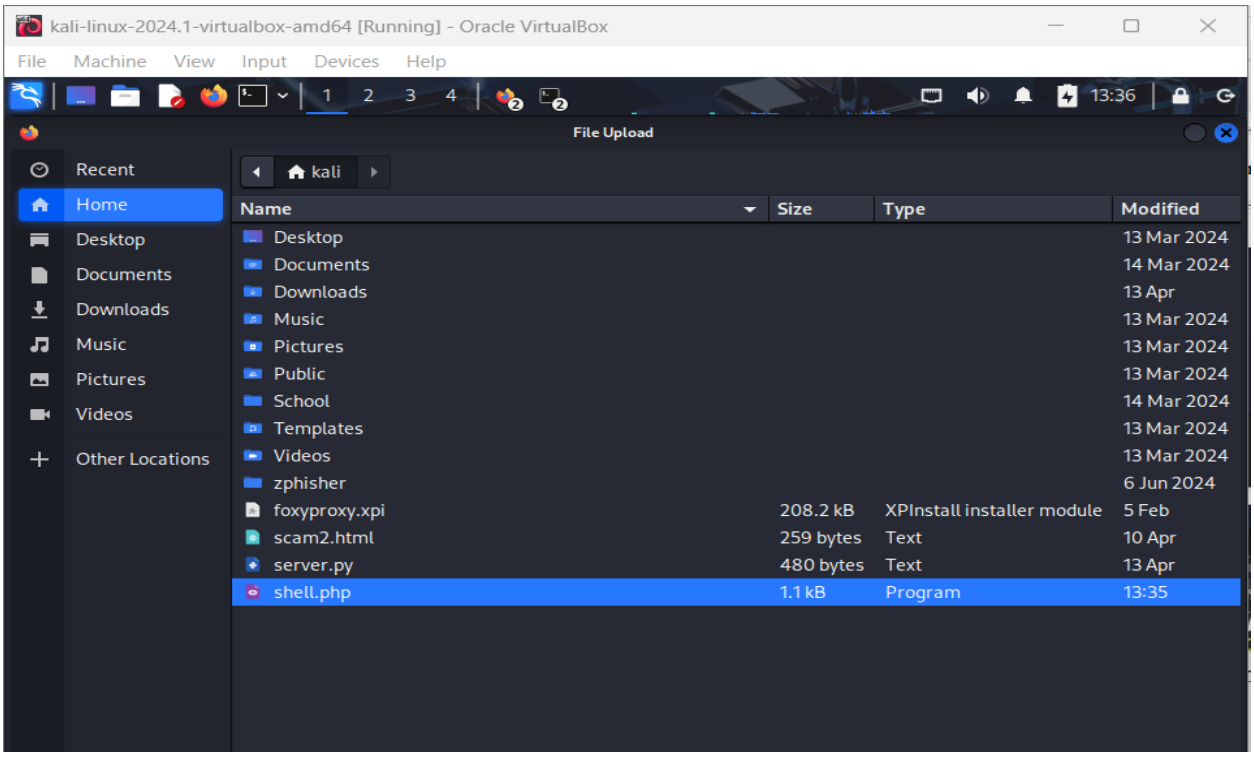

Figure 44:Navigating to the file.

Uploaded file successfully.

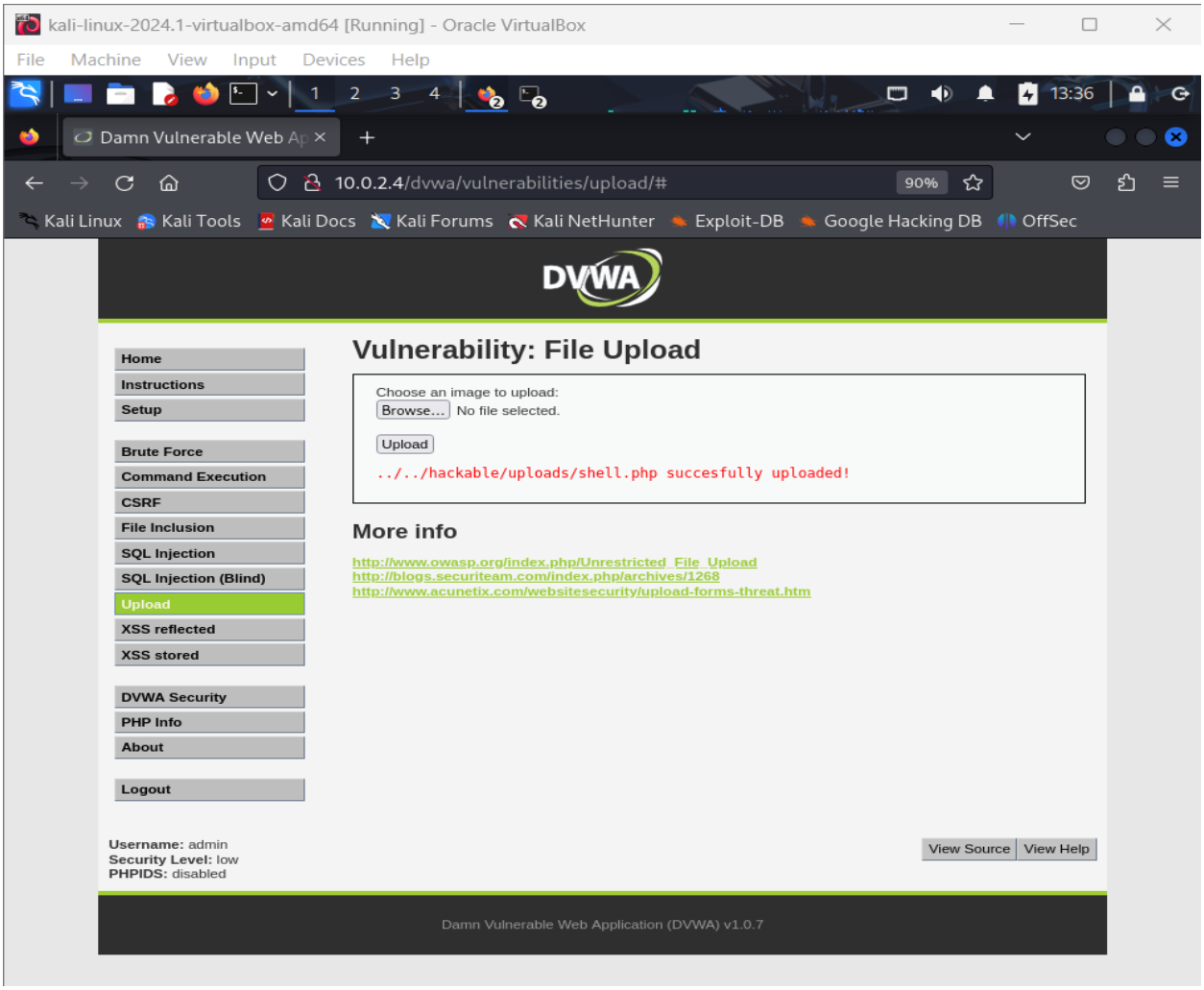

Figure 45:Successfully uploaded.

Crafting a malicious payload.

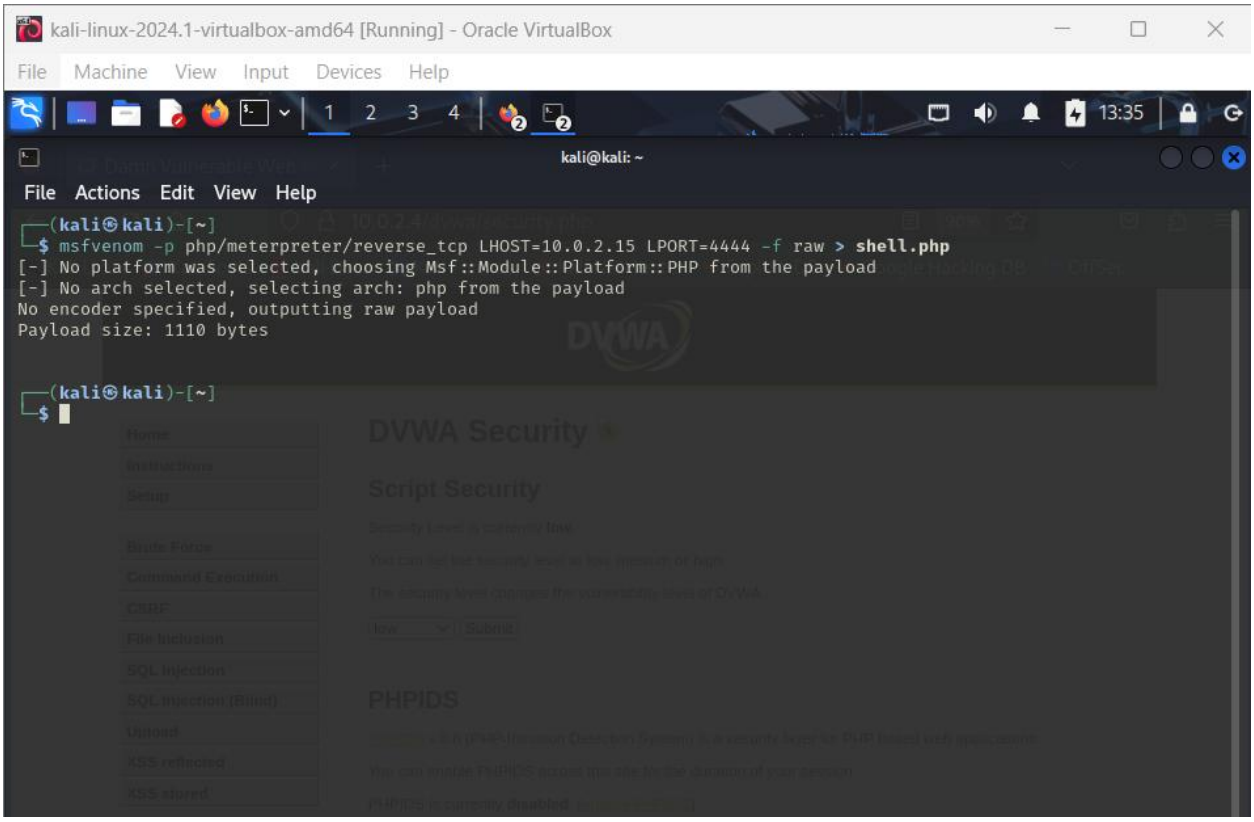

Figure 46:Crafting a malicious payload.

Set up a Listener in Kali Linux.

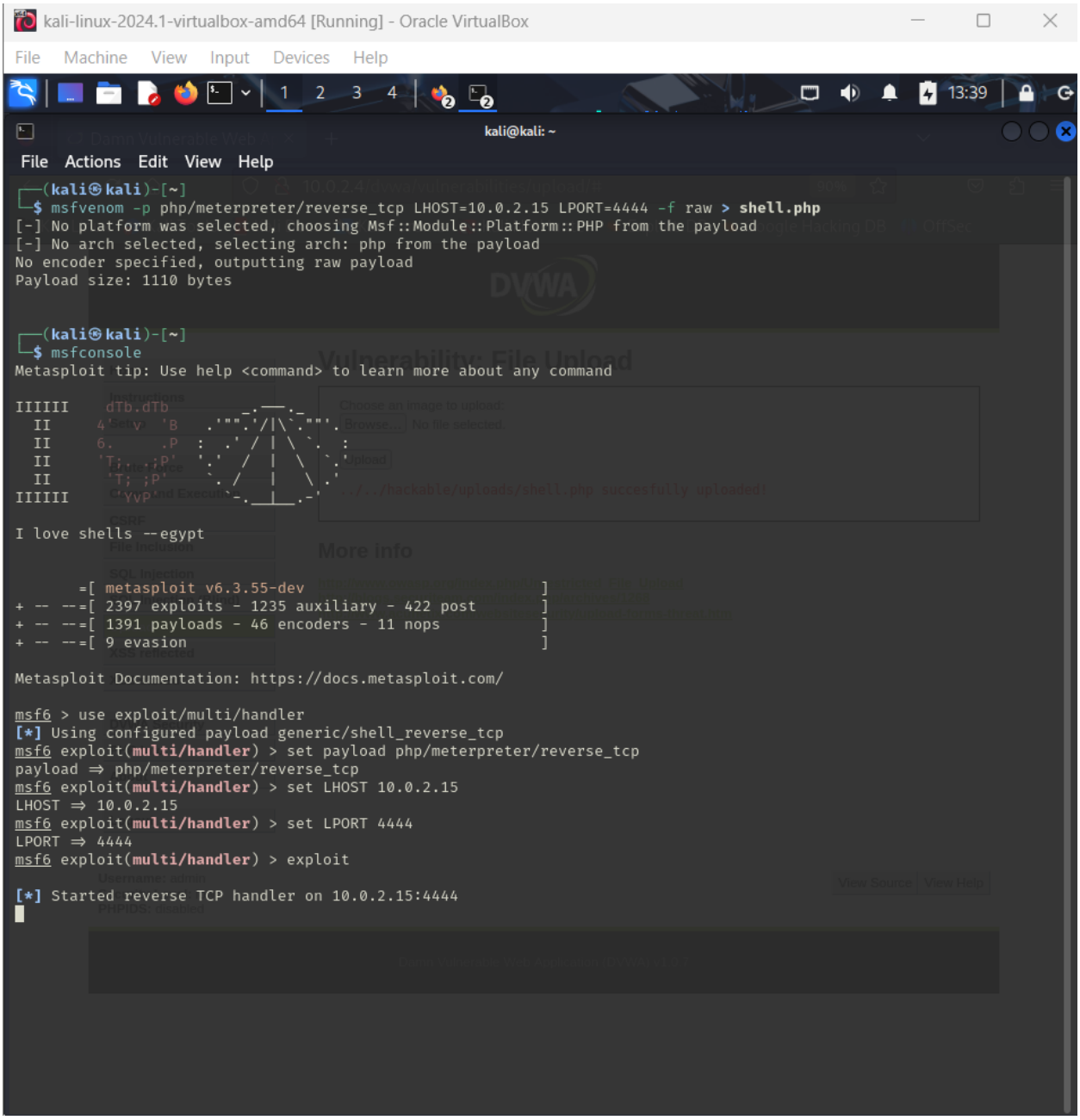

Figure 47:Set up a Listener in Kali Linux.

Trigger the Payload and check for an active meterpreter session.

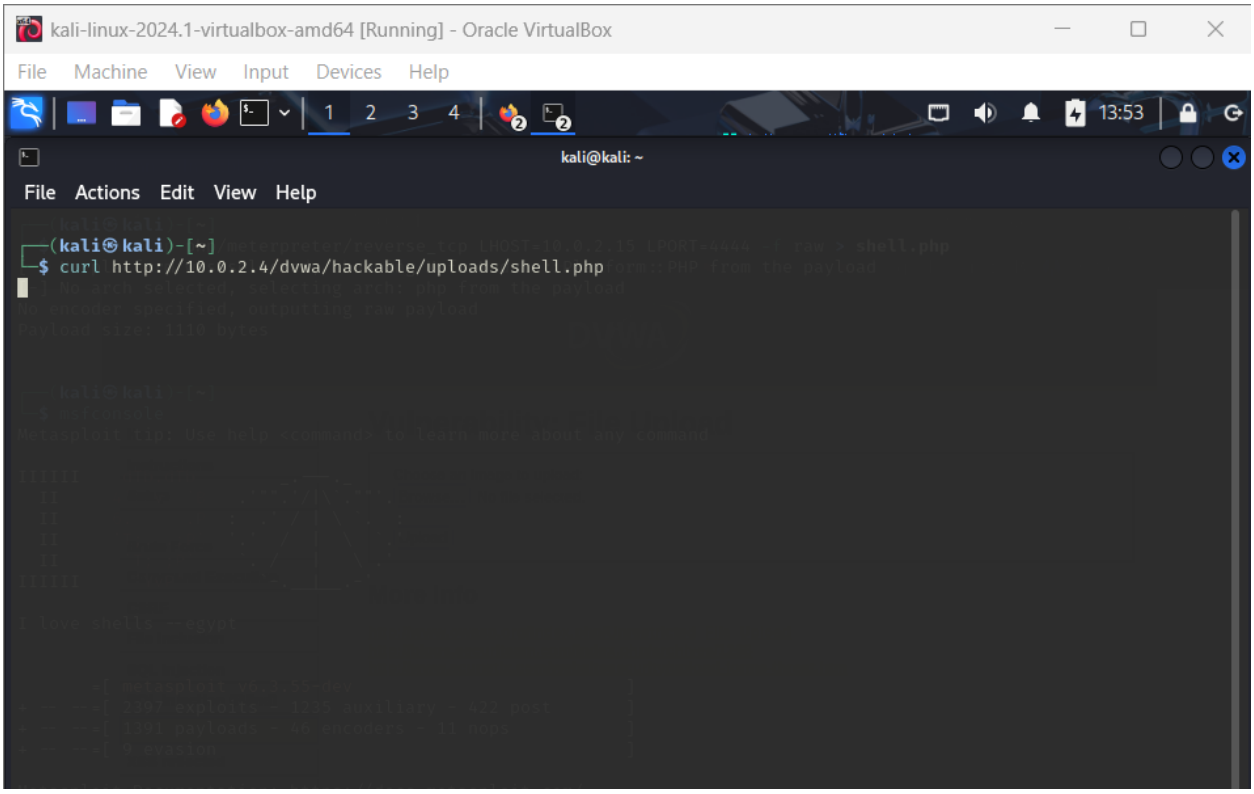

Figure 48:Trigger the Payload and check for an active meterpreter session.

The information about computer, OS and meterpreter is shown.

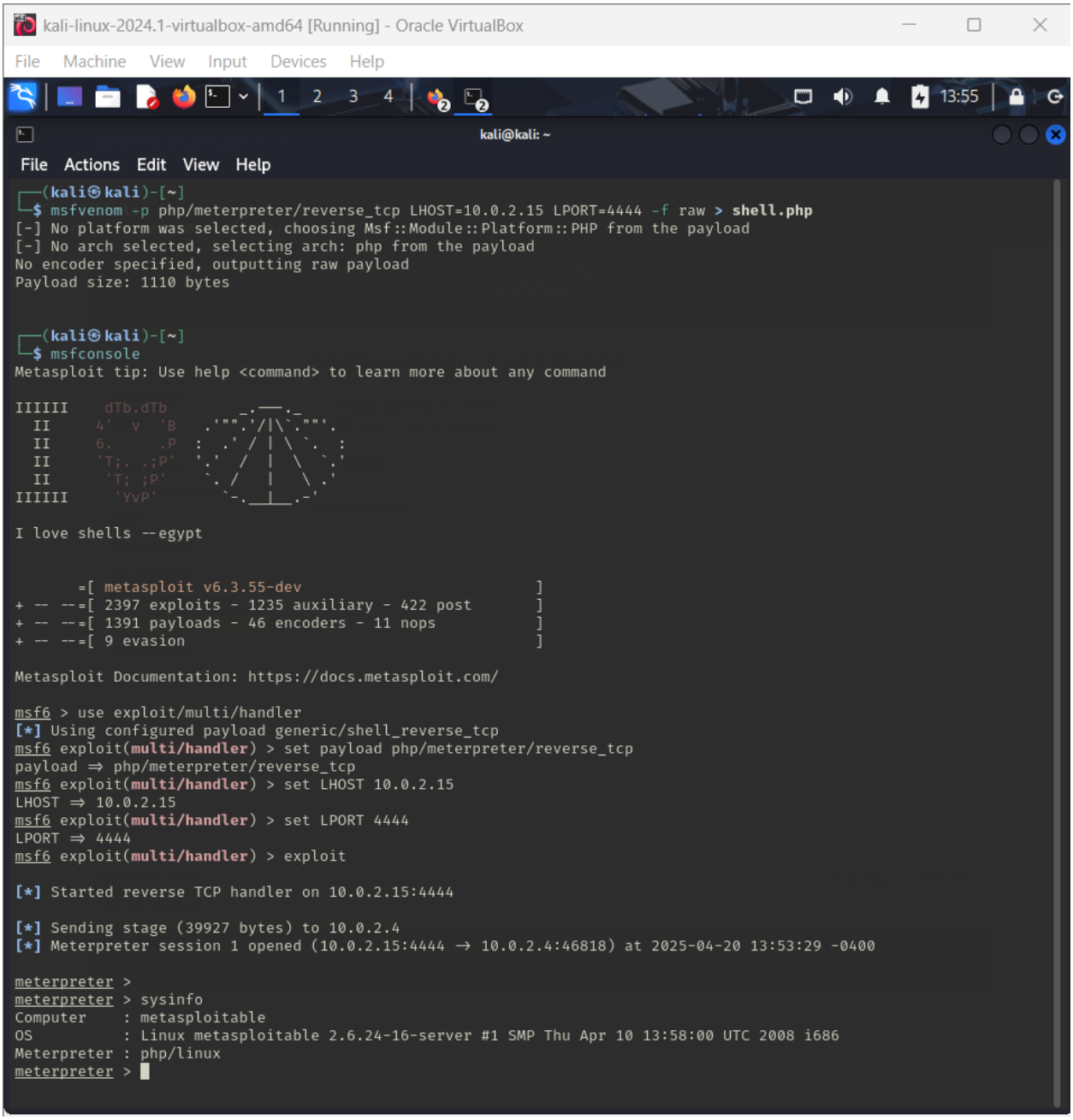

Figure 49:The information about computer, OS and meterpreter is shown.

f. Reporting:

Penetration testing was done in a number of steps starting with intelligence gathering, where DVWA information was obtained using a web browser, and IP addresses for both Metasploitable 2 and Kali Linux were obtained using (ifconfig). Port scanning using Nmap revealed FTP, SSH, and HTTP open ports. In threat modeling, a Nessus scan of Metasploitable 2 identified 59 vulnerabilities, including critical and high-risk vulnerabilities and HTTP and SSL-specific vulnerabilities. In exploitation, Burp Suite was used with proxy setup and imported CA certificate to intercept traffic; DVWA's security level was set to low and medium in varying stages to allow brute-force attacks with custom wordlists created using Crunch. Valid credentials were obtained comparing lengths of response content, which provided access to the web application. A file upload vulnerability was exploited during post-exploitation to upload a malicious payload, which upon execution established a reverse shell via Meterpreter. This provided complete system access, revealing sensitive system details and indicating exploitation of the target to be successful.

4. MITIGATION

During the demonstration, brute-force attacks were carried out using three different ways to show how attackers might try to guess passwords. The first method tested a few simple login attempts by typing common passwords manually which is called a low-level attack. The second method used tools like Burp Suite and Hydra which can automatically send many login attempts quickly which showed a medium-level attack. And the third method involved performing a dictionary attack using a list of common passwords to attempt a large number of guesses in a short time which demonstrated a high-level brute-force attack. These demonstrations helped us to understand how the level of risk changes based on the method used.

Based on the observations made during the demonstration, here are some useful ways to prevent or reduce brute-force attacks:

1.Account Lockout Policy

This means locking a user's account after several wrong password attempts. For Example: After 5 failed tries the system can lock the account for 15 minutes or until someone unlocks it manually. This strategy stops the attacker from trying over and over again. This method is best for low-level attacks with only a few login attempts. (Esheridan, 2022)

2.CAPTCHA or reCAPTCHA

CAPTCHA adds a small test on the login page, like typing letters or choosing images to check if the user is a real person. Bots and brute-force tools cannot solve these easily, so it blocks them from continuing their attack. This method is usually best for Medium-level attacks which is done using automated tools. (Esheridan, 2022)

3.Rate Limiting/Request Throttling

This method limits how many times someone can try to login in a short time. If too many attempts are made too quickly then the system blocks further tries for a while. This helps slow down or stop brute-force attack. This method is best for Medium to high-level attacks with fast login attempts. (Thakur, 2023)

4.Multi-Factor Authentication (MFA)

MFA adds an extra step to the login process like entering a code to a phone or email. Even if the attacker guesses the password, they can't login without the second code. This method is best for High-level attacks where the passwords have been cracked. (Holdsworth & Kosinski, 2024)

5.Strong Password Policies

This means users are required to create passwords that are hard to guess. For example: Passwords with uppercase and lowercase letters, numbers and symbols. This makes it more difficult for brute-force tools to guess the password. This method is best for preventing low-level attacks using simple or common passwords. (Ruchini, 2023)

6. Logging and Real-Time Monitoring

A system manages login activity by triggering alerts whenever many failed login attempts exceed unexpected levels. The monitoring system enables prompt responses to stop upcoming attacks. Professionals use all levels of brute-force attacks through this technique. (Bowman, 2023)

5.EVALUATION

Every Method used to stop brute-force attacks has it's own strengths and weaknesses. Some Work better for low-level attacks while others are needed for more advanced attacks. To decide which technique to use, it's important to look at both the positive and negative sides if each. This helps to choose the most suitable solutions depending on the situation and level of risk.

Here are some of the advantages and disadvantages of each technique:

1.Account Lockout Policy

Advantages

- **Stops repeated login attempts:** It locks the account after a few wrong tries which can block attackers from continuing to guess passwords.
- **Easy to set up:** The implementation of this method becomes very simple since major login platforms support its deployment.

- **Good for small systems:** Small systems easily manage lockouts because they do not cause disruption to many users.
- **Alerts security teams:** Locking accounts can alert admins so that they can check if something suspicious is happening in the system.
- **Works well with monitoring tools:** Monitoring tools and log alerts work together to provide enhanced security protection when employed with Brute-Force protection Methods.

Disadvantages

- **Can be misused by attackers**: Hackers can intentionally try wrong passwords to lock other users accounts, causing trouble or denial of service(D0S).
- **Can reveal valid usernames:** Only real usernames can be locked so attackers may find out which accounts exist by testing for lockouts.
- **Can lock accounts again and again:** Even after the account is unlocked, attackers can quickly lock it again making it hard for users to use their account.
- **Doesn't Stop slow attacks:** This method fails to prevent gradual password guessing because extended time between passwords attempts results in no account blocking.
- **Might block users during important moments:** When users try to perform urgent operations online such as money transfers or rapid purchases through internet shopping platforms being temporarily blocked could prevent them from successfully completing their tasks.

2. CAPTCHA OR reCAPTCHA

Advantages

- **Blocks bots and Scripts:** CAPTCHA operators test if the login request is from a real human or a computer. This blocks bots from conducting multiple password attempts because it blocks their access.
- **Hard for computers to guess:** The attack process fails when bots cannot pass CAPTCHA or reCAPTCHA tests.
- **Simple for people to use:** Most CAPTCHAs are easy for humans to solve like choosing pictures or typing letters.

- **Improves Overall Security:** The implementation of CAPTCHAs as a security measure can lead to lower attack rates through their basic verification process when even combined with lockout accounts or Multi-Factor Authentication (MFA).
- **Easy to add websites:** Websites can easily combine free CAPTCHA protection through reCAPTCHA services on their login pages forms and registration systems.

Disadvantages

- **Can be frustrating for users:** Users experience frustration when trying to see the CAPTCHA elements since some letters or pictures become difficult to figure out.
- **Not user-friendly for everyone:** Solving CAPTCHAs poses challenges to users who have disabilities or who belong to the older population.
- **Smart bots can sometimes solve them:** Advanced bots or services may be able to bypass simple CAPTCHAs especially if they're not updated.
- **Slows down user login: CAPTCHAs** create an additional step during login that delays the process of accessing the system**.**
- **Needs to be well-designed:** A poorly designed CAPTCHA system makes bots susceptible to guessing correct answers. Real users might not resolve difficult CAPTCHAs which can lead them to fail in the process.

3. Rate Limiting/Request Throttling

Advantages

- **Slows down attackers:** This method protects systems by slowing down attackers who try to conduct multiple passwords attempts too quickly.
- **Protects the systems:** It safeguards the systems by stopping servers from becoming overloaded due to an excess of simultaneous login requests.
- **Easy to set up:** Many websites and security tools have rate limiting and you can turn it on easily using basic settings or add-ons.
- **Works with other protections:** It provides enhanced protection effectiveness. When used together with account lockout and CAPTCHA and MFA.
- **Reduces abuse:** The security system eliminates all types of abuse including brute-force attacks while it additionally blocks spam and bot traffic.

Disadvantages

- **Might block real users:** Real users may occasionally get blocked due to incorrect password recovery attempts.
- **Can be bypassed using many Ips**: Attackers can avoid rate limiting by sending requests from multiple IP addresses.
- **Needs careful setup:** Strict setup is essential because inappropriate configurations may result in system weaknesses.
- **Could slow down normal activity:** The system might run slower than normal when users access it because the specified rate limitations might be too restrictive for everyone including regular users.
- **Needs monitoring:** System administrators need to regularly check on these settings because their proper functioning is very essential.

4. Multi-Factor Authentication (MFA)

Advantages:

- **Adds extra security:** Even if someone steals the password, they still can't login without the second code.
- **Protects from being hacked:** MFA is very effective in stopping attackers from breaking in.
- **Used in Many Websites and apps:** Many services support MFA and users are used to using it.
- **Uses two steps to confirm identity:** It checks both something you know(password) and something you have(like a phone).

Disadvantages

- **You need your phone or email:** If the user doesn't have access to their device, they can't log in.
- **It can be hard to recover if lost:** If someone loses their phone or access to email then it may be hard to get back into their account.

- **Older systems may not support:** Some of the old websites or apps needs updates to use MFA.
- **It takes time to log in:** It adds one extra step during login.
- **Some users may get confused:** Not everyone understands MFA well so they might need help.

5. Strong Password Policies

Advantages

- **Makes passwords harder to guess:** A strong password is more secure and an attacker will find it more difficult to crack a strong password.
- **Blocks weak and common passwords:** It stops users from using simple and common passwords that are easy to figure out like “123456” or “password”.
- **Teaches good habits:** It motivates users to come up with better and more secure passwords.

Disadvantages

- **Can be hard to remember**: Users may become confused by strong passwords because they are more difficult to remember.
- **Might lead to reuse:** It is too risky for users to use the same passwords across multiple websites.
- **Not enough on it’s own:** Strong passwords are best, but they work more better when it is combined with other protections like multi-factor authentication.

6. Logging and Real-time Monitoring

Advantages

- **Spots attacks quickly:** It alerts admins when there are too many failed logins or other unusual actions.
- **Works with other methods:** Monitoring supports other protections like account lockout and CAPTCHA by giving live updates.
- **Helps investigate attacks**: Logging helps in tracking how and when an attack happens which makes easier in fixing security gaps.

- **Improves Overall Security:** Continuously monitoring the system helps to keep the system safe by reacting in real time.

Disadvantages:

- **Requires tools and setup:** users need some special software or platform to collect and monitor logs.
- **Can create too many alerts:** If it is not set up properly then it may send too many false warnings which may create difficulty in managing.
- **Takes time to review:** Analyzing logs can take time especially in large systems with large amounts of data.
- **Needs human-attention:** Someone must regularly check alerts and logs to catch threats early.

5.1 Application Areas

The techniques which are used to stop brute-force attacks are useful in many real-life systems. These methods helps to protect user data and prevent attackers from getting into accounts. Below are some areas where these security techniques are commonly used:

1.Online banking systems

Banking websites are a major target for the attackers who try to break into user accounts to steal money or personal information. For example: Many banks now use multi-factor authentication (MFA) which requires a second code after entering the password. If someone tries to guess the password multiple times then the account will get locked. Real-time monitoring also alerts the bank if someone tries to log in many times from different locations.

2. E-Commerce Websites

Online Stores like Amazon, Daraz store customer information including addresses and payment details. To protect user accounts these websites use CAPTCHA and rate limiting to stop bots from trying to guess passwords. If a user tries too many times then the system blocks further attempts for a period of time.

3. Employee Login Systems in Companies

Companies use internal systems where employees log in daily to access files or work portals. If an attacker tries to guess employee passwords, then the features like account lockout and real-time alerts help stop the attack quickly. Many companies also require employees to use MFA to add extra protection.

4.Social-Media platforms

Sites like Facebook, Instagram and Twitter use CAPTCHA and MFA to top attackers from taking over accounts. If someone enters the wrong password too many times, then the system blocks the login and sends a notification to the user. These steps help keep social media accounts safe from brute-force attacks.

5.Email Services

Popular email platforms like Gmail and Outlook use multiple security layers to prevent brute-force attacks. If someone tries to guess the password many times then the account is temporarily locked. These platforms also monitor login attempts and send alerts to the account owner if an unknown device tries to log in.

6. References

1k, g., 2024. *Kali.* [Online]
Available at: https://www.kali.org/docs/introduction/what-is-kali-linux/
[Accessed 01 03 2025].

Bowman, K., 2023. *Pathlock.* [Online]
Available at: https://pathlock.com/learn/real-time-monitoring/
[Accessed 21 04 2025].

Christian, A.-A., 2024. *SSL2BUY.* [Online]
Available at: https://www.ssl2buy.com/wiki/brute-force-attack
[Accessed 31 03 2025].

Esheridan, 2022. *owasp.* [Online]
Available at: https://owasp.org/www-community/controls/Blocking_Brute_Force_Attacks
[Accessed 22 04 2025].

Goodman, C., 2025. *Balbix.* [Online]
Available at: https://www.balbix.com/insights/what-is-a-vulnerability/
[Accessed 21 04 2025].

Heyman, T. M., 2024. *Criipto.* [Online]
Available at: https://www.criipto.com/blog/what-is-authentication
[Accessed 21 04 2025].

Holdsworth, J. & Kosinski, M., 2024. *IBM.* [Online]
Available at: https://www.ibm.com/think/topics/multi-factor-authentication
[Accessed 20 04 2025].

Larson, J., 2024. The Dark Truth about Cyber Security. *Marriott Student Review,* 7(2).

Lenaerts-Bergmans, B., 2022. *CrowdStrike.* [Online]
Available at: https://www.crowdstrike.com/en-us/cybersecurity-101/cyberattacks/password-spraying/
[Accessed 1 03 2025].

Martinez, J., 2024. *Strongdm.* [Online]
Available at: https://www.strongdm.com/blog/brute-force-attack
[Accessed 20 04 2025].

Mohammed, Bello Suleiman1 ; Romanus, Robinson2, 2024. International Journal of Innovative Science and Research Technology. *Long-Short Term Memory Network Based Model for,* 9(7), p. 20.

O'Sullivan, F., 2024. *Proton.* [Online]
Available at: https://proton.me/blog/what-is-dictionary-attack
[Accessed 21 04 2025].

Raza, M., 2025. *Splunk.* [Online]
Available at: https://www.splunk.com/en_us/blog/learn/cybersecurity-exploits.html
[Accessed 21 04 2025].

Ridwan, R., 2024. Using the Penetration Testing Execution Standard Method (PTES) for Wireless Network Security Analysis. *Greenation Computer and Information Review*, 1(1), pp. 25-32.

Ruchini, C., 2023. *Medium.* [Online]
Available at: https://medium.com/identity-beyond-borders/the-importance-of-implementing-a-strong-password-policy-bb08a9c7b475
[Accessed 21 04 2025].

S3Curiosity, 2023. *Medium.* [Online]
Available at: https://medium.com/@S3Curiosity/exploring-dvwa-a-valuable-tool-for-ethical-hacking-0e9249e54f43
[Accessed 21 04 2025].

Sahoo, P. K., 2025. *Qualysec.* [Online]
Available at: https://qualysec.com/penetration-testing-execution-standard/
[Accessed 21 04 2025].

Sahu, V., 2024. *Scaler Topics.* [Online]
Available at: https://www.scaler.com/topics/cyber-security/burp-suite/
[Accessed 21 04 2025].

Scott, M. O., 2024. *PingIdentity.* [Online]
Available at: https://www.pingidentity.com/en/resources/blog/post/what-is-brute-force-attack.html
[Accessed 27 03 2025].

Shivanandhan, M., 2020. *freeCodeCamp.* [Online]
Available at: https://www.freecodecamp.org/news/what-is-nmap-and-how-to-use-it-a-tutorial-for-the-greatest-scanning-tool-of-all-time/
[Accessed 01 03 2025].

Staff, C., 2025. *Coursera.* [Online]
Available at: https://www.coursera.org/articles/types-of-cyber-attacks
[Accessed 03 03 2025].

Swathi, K., 2022. Brute Force Attack on Real World Passwords. *International Journal of Research Publication and Reviews,* 3(11), pp. 552-558.

Thakur, G. k., 2023. *Medium.* [Online]
Available at: https://medium.com/@gopalkthakur/throttling-or-rate-limiting-41feb9700275
[Accessed 20 04 2025].