Vulnerability Report on Mr. Robot Machine

Objective:

The goal of this penetration test is to brute-force access to a web application on the Mr. Robot machine This will involve identifying an entry point, such as a login form, and applying a brute-force attack to retrieve credentials.

Tools Used:

- **Nmap**: For network scanning and service discovery.
- **Gobuster/Dirbuster**: For directory brute-forcing.
- **Hydra**: For brute-forcing web forms (login).
- **Wordlists**: Common wordlists like rockyou.txt for password brute-forcing.

Steps to Perform the Attack:

Step 1: Initial Scanning and Reconnaissance

Before starting the brute-force attack, I performed an initial scan to gather information about the machine's IP and the open ports.

1. **Network Scan using Nmap**: Run the following command to discover open ports and services running on the target machine:

nmap -A <target_ip>

The scan revealed the following key services:

    - Port 80: HTTP (Apache web server)
    - Port 443: HTTPS
    - Port 22: SSH

Step 2: Web Application Reconnaissance

After identifying an active web server on port 80, I accessed the website and found a CMS-based interface with potential login fields.

1. **Directory Brute-Forcing with Gobuster**: To find hidden directories and files on the web server, I used Gobuster:

gobuster dir -u http://<target_ip> -w /usr/share/wordlists/dirb/common.txt

This revealed several interesting directories, including /robots.txt, which contained sensitive information.

2. **Robots.txt Analysis**: Accessing http://<target_ip>/robots.txt, I found a clue pointing to a wordlist (fsocity.dic). This file can be used later for password brute-forcing.

Step 3: Brute-Forcing the Login Page

With the gathered information, I proceeded to brute-force the login credentials using the wordlist found earlier.

1. **Identifying the Login Page**: After exploring the directories, I found a login form. Since no credentials were immediately available, brute-forcing was necessary.
2. **Using Hydra for Brute-Force Attack**: Hydra is a powerful tool for login brute-forcing. I used the following command to perform the attack:

hydra -l <username> -P /path/to/fsocity.dic <target_ip> http-post-form "/login:username=^USER^&password=^PASS^:F=incorrect"

   - -l <username>: Specify the username (either a known or guessed username).
   - -P /path/to/fsocity.dic: Use the wordlist fsocity.dic for passwords.
   - http-post-form: Target the login form with the format of the POST request and error string.

After running Hydra, it successfully found the correct password for the given username.

Step 4: Gaining Access

With the username and password obtained from Hydra, I was able to log into the web application.

1. **Further Enumeration**: Once inside the system, further privilege escalation techniques could be used, but the focus of this report is on the brute-force attack.

Mitigation:

To prevent brute-force attacks like this, the following security measures should be implemented:

- Implement account lockout after a certain number of failed login attempts.
- Use CAPTCHAs to prevent automated login attempts.
- Enforce strong password policies.

- Regularly monitor logs for suspicious activity.

Conclusion:

The brute-force attack was successfully executed using **Hydra** with the fsocity.dic wordlist, leading to a successful login on the Mr. Robot machine. The vulnerability existed due to a lack of account lockout mechanisms and weak passwords. By improving login security, such attacks can be mitigated in the future.