Badly written web applications are extremely vulnerable. It is expected by 2021, the loss due to cybercrimes could be $6 trillion. This includes, damage and destruction of data, stolen money, lost productivity, theft of intellectual property, theft of personal and financial data, embezzlement, fraud, post-attack disruption to the normal course of business, forensic investigation, restoration and deletion of hacked data and systems, and reputational harm. Substantial chunk of these losses could come from hacked web applications.

Analysis of web applications have shown that in 2017, nearly 50% of web applications were at the risk of unauthorised access. This number rose to nearly 72% by 2018. In fact as much as 19% of the web applications had critical vulnerabilities that allowed a hacker to take control of the application and the operating system.

Over this week we will study` some of these vulnerabilities and understand how they can be exploited by the hacker. We will use Kali Linux and Damn Vulnerable Web Application (DVWA) for this purpose.

If you have Metasploitable 2 installed then you already have DVWA running at port 80 on that VM. Unfortunately it is an older version and we would be better off using the most recent version. We will install the latest version on our Kali Linux VM. We could have upgraded the old version running on Metasploitable 2 VM, but multiple VM slow down the system. Some of our attacks are computationally intensive, so using a single VM would be more efficient.

DVWA is written in PHP and requires Apache web server along with MySQL database server. I followed following installation process. All the steps need to be performed as superuser.

Install Apache - **$ apt-get install apache2**

All the web application files would be located under /var/www/html (webroot for Apache2)

Install DVWA - **$ cd /var/www/html**

$ apt-get install git

$ git clone https://github.com/ethicalhack3r/DVWA.git

The DVWA will get installed in /var/www/html/DVWA

$ chmod –R 777 DVWA

This will make DVWA and its sub directories world readable. For a real web application this would be a really bad idea and should never be done. Actual access permissions should be kept to minimum and access to various directories should be managed through Apache2 configuration files.

Install MySQL - Unfortunately due to licencing issues we will be forced to use a fork of MySQL called MariaDB

$ apt-get install mariadb-server

Start the servers -

$ service apache2 start or systemctl start apache2

$ service mariadb start or systemctl start mariadb

Install PHP - I found that PHP 7.3 was already installed on my system. Following packages were also already installed on my Kali Linux systems.

1. php-mysql
2. php-gd

If you don't have these package, please install them using apt-get.

Configure DVWA

$ cd /var/www/html/DVWA/config

$ cp config.inc.php.dist config.inc.php

Open *config.inc.php* using any text editor. Locate following lines

$_DVWA['db_database'] = 'dvwa';

$_DVWA['db_user'] = 'root';

$_DVWA['db_password'] = 'p@ssw0rd';

This configuration works with MYSQL. We are using MariaDB, so some changes are necessary.

MariaDB *root* user has no password. To access database from command prompt, use following command.

$ mysql –u root -p

When prompted for MariaDB *root* user password, just press Enter key.

At the database prompt type following commands

mysql> **create database dvwa;**

Query OK, 1 row affected (0.00 sec)

mysql> **grant all on dvwa.* to dvwa@localhost identified by 'nopass';**

Query OK, 0 rows affected, 1 warning (0.01 sec)

mysql> **flush privileges;**

Query OK, 0 rows affected (0.00 sec)

Now change the entries in the *config.inc.php* as follows

$_DVWA['db_database'] = 'dvwa';

$_DVWA['db_user']   = 'dvwa';

$_DVWA['db_password'] = 'nopass';

Save the file and exit.

Configure PHP

$ cd /etc/php/7.3/apache2/

Edit *php.ini* file using any text editor

Locate and enable following settings

allow_url_fopen = On

allow_url_include = On

Save the file and exit.

Restart Apache and MariaDB

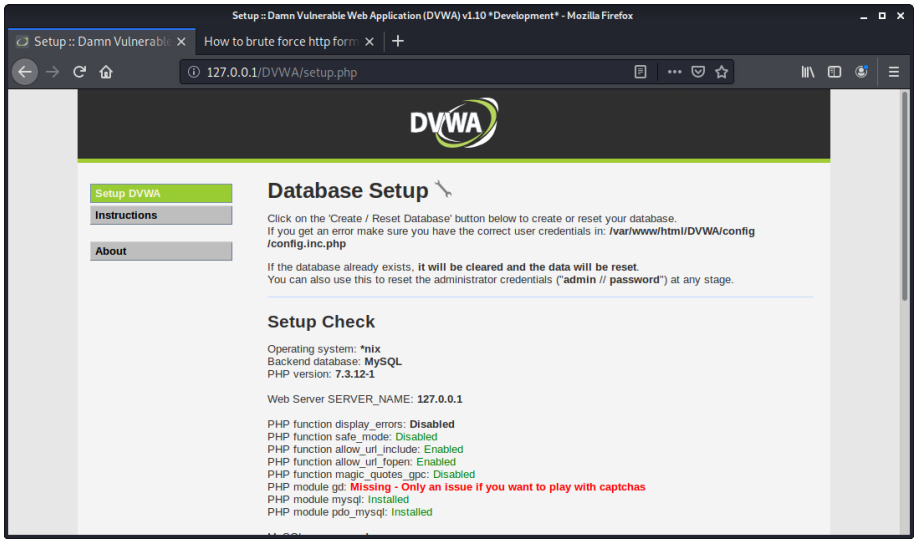
$ service apache2 restart

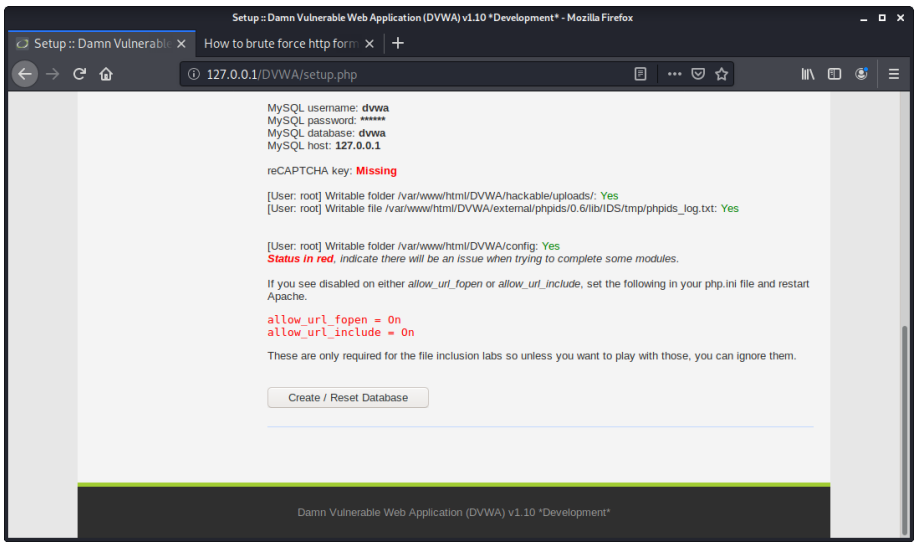
$ service mariadb restart

Start the browser and browse to following url

http://127.0.0.1/DVWA/

Follow the instructions on the page. Create the database/tables by clicking on create/reset button. Ignore any error related to Captcha. We won't need that functionality.

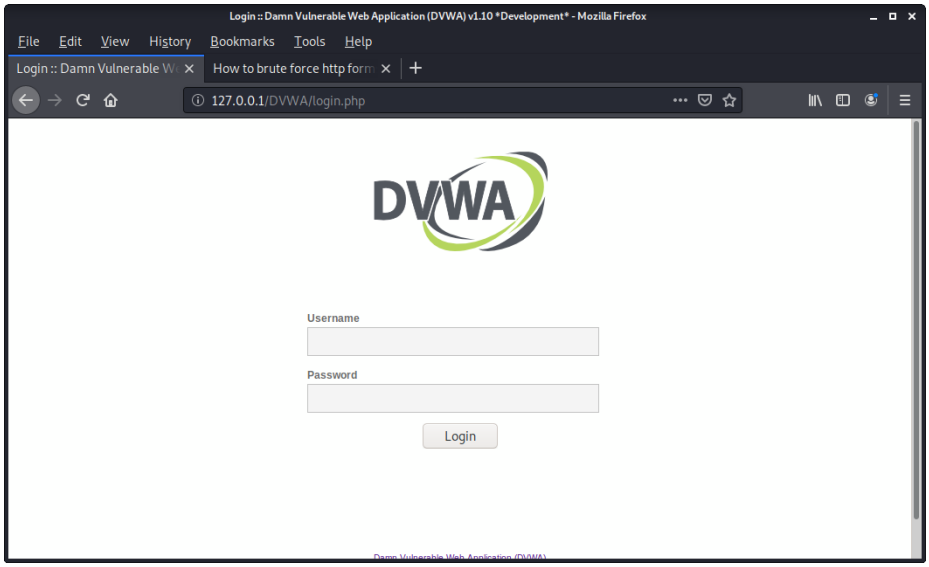




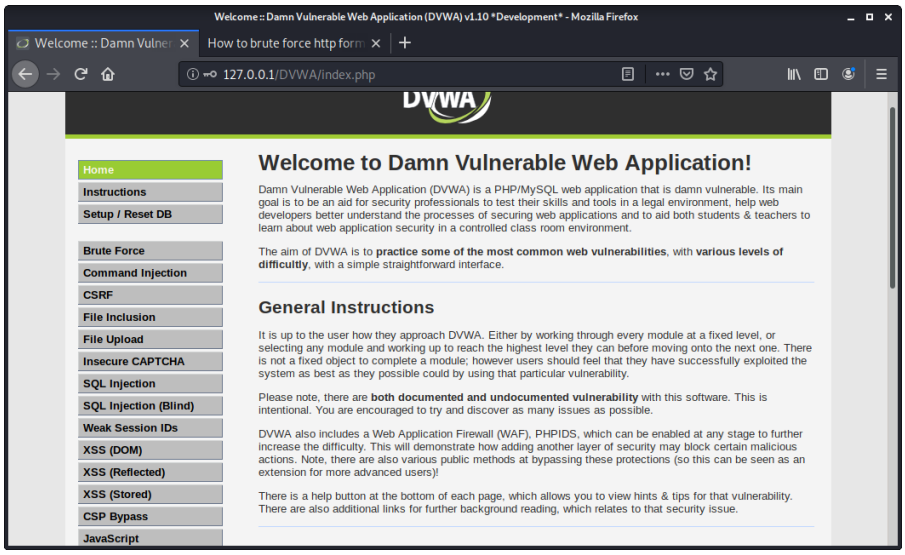
Now you will be presented with login page

Login id : admin

Password: password



Log into the DVWA application.



On the left hand side, you can see links to all the vulnerabilities that can be exploited.