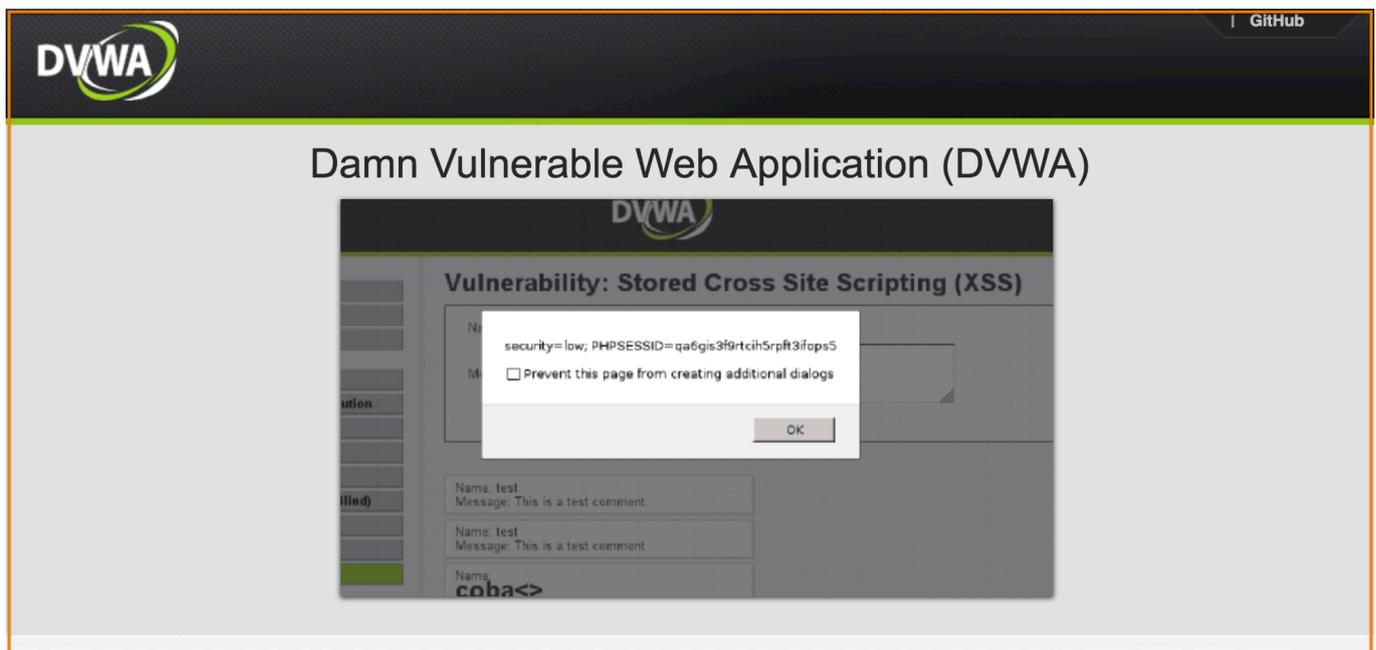


Create Your First Ethical Hacking Environment with DVWA

In this tutorial, I'll demonstrate you to setup Damn Vulnerable Web Application (DVWA) along with Apache, MySQL, PHP on localhost. It's always been a concern for newbies that where they should practice and explore the vulnerabilities.

If you are one of those guys, DVWA would be for you to figure it out yourself. I will help you create a hacking environment into your Linux distro to practice and test your skills.

And, if you are new to Web Security and want to get ahead in the field of CyberSecurity. I'd suggest that you read '[How to Become a Web Security Researcher](#)', you'll richly benefit from the tips and resources provided.



Attacks Covered in DVWA

Brute Force
Command Execution
CSRF
File Inclusion
SQL Injection
SQL Injection (Blind)
Shell Uploading
XSS (Reflected)
XSS (Stored)

Prerequisites:

Virtual Machine: [VirtualBox](#)

Linux Distro: [Kali Linux](#), [Parrot Security OS](#) or [elementaryOS](#) (or any other [Linux Distribution](#))

I prefer using elementaryOS which is Lightweight Linux based distribution, but you can follow the same instructions for Kali Linux which is aimed at advanced Penetration Testing and Security Auditing.

Remember, we need to use a virtual machine and not a connected server because DVWA is really vulnerable and should only be installed on your virtual machine with NAT.

Setup & Install DVWA Into Your Linux Distribution

DVWA is made with PHP and MySQL for security professionals or aspiring security professionals to discover as many issues as possible and exploit some of the most commons vulnerabilities of web platforms like SQL injection, Cross Site Scripting

(XSS), Cross Site Request Forgery (CSRF), and more.

Note: This guide is for beginners. If you're unable to complete any of the steps or encounter any error message during the installation. I encourage you to use [StackOverflow](#) for an answer or leave a comment below.

Step 1. Setup Web server (Install Apache)

To install Apache, Open your Terminal and type the following:

```
sudo apt install apache2
```

Once done, type 127.0.0.1 in the browser and you will see the default Apache 2 web page, similar to this:



Apache2 Debian Default Page

debian

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Debian systems. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

Configuration Overview

Debian's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Debian tools. The configuration system is **fully documented in `/usr/share/doc/apache2/README.Debian.gz`**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the `apache2-doc` package was installed on this server.

The configuration layout for an Apache2 web server installation on Debian systems is as follows:

```
/etc/apache2/  
|-- apache2.conf  
|   |-- ports.conf  
|-- mods-enabled  
|   |-- *.load  
|   |-- *.conf  
|-- conf-enabled  
|   |-- *.conf  
|-- sites-enabled  
|   |-- *.conf  
|
```

- `apache2.conf` is the main configuration file. It puts the pieces together by including all remaining configuration files when starting up the web server.
- `ports.conf` is always included from the main configuration file. It is used to determine the listening ports for incoming connections, and this file can be customized anytime.
- Configuration files in the `mods-enabled/`, `conf-enabled/` and `sites-enabled/` directories contain particular configuration snippets which manage modules, global configuration fragments, or virtual host configurations, respectively.
- They are activated by symlinking available configuration files from their respective `*-available/` counterparts. These should be managed by using our helpers `a2enmod`, `a2dismod`, `a2ensite`, `a2dissite`, and `a2enconf`, `a2disconf`. See their respective man pages for detailed information.
- The binary is called `apache2`. Due to the use of environment variables, in the default configuration, `apache2` needs to be started/stopped with `/etc/init.d/apache2` or `apache2ctl`. **Calling `/usr/bin/apache2` directly will not work** with the default configuration.

If you see this page, then congratulations — you have successfully installed Apache.

When you are done looking at this test page, you can remove it by typing the following command:

```
sudo rm /var/www/html/info.html
```

Step 2. Download DVWA

We need to download the archive of DVWA from Github.

To install Git, type following command:

```
sudo apt-get install git
```

Go to the apache2 folder.

```
cd /var/www/html/
```

Clone DVWA from Github, type the following command:

```
sudo git clone https://github.com/ethicalhack3r/DVWA.git
```

Once done, type `127.0.0.1/DVWA/` in the browser and you will see the DVWA page, similar to this:

```
← → ↻ ⓘ 127.0.0.1/DVWA/ ☆ ...
<?php
define( 'DVWA_WEB_PAGE_TO_ROOT', '' );
require_once DVWA_WEB_PAGE_TO_ROOT . 'dvwa/includes/dvwaPage.inc.php';

dvwaPageStartup( array( 'authenticated', 'phpids' ) );

$page = dvwaPageNewGrab();
$page[ 'title' ] = 'Welcome' . $page[ 'title_separator' ].$page[ 'title' ];
$page[ 'page_id' ] = 'home';

$page[ 'body' ] .= "
<div class=\"body_padded\">
  <h1>Welcome to Damn Vulnerable Web Application!</h1>
  <p>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.</p>
  <p>The aim of DVWA is to <em>practice some of the most common web vulnerabilities</em>, with <em>various levels of difficulty</em>, with a simple straightforward interface.</p>
  <hr />
  <br />

  <h2>General Instructions</h2>
  <p>It is up to the user how they approach DVWA. Either by working through every module at a fixed level, or selecting any module and working up to reach the highest level they can before moving onto the next one. There is not a fixed object to complete a module; however users should feel that they have successfully exploited the system as best as they possibly could by using that particular vulnerability.</p>
  <p>Please note, there are <em>both documented and undocumented vulnerability</em> with this software. This is intentional. You are encouraged to try and discover as many issues as possible.</p>
  <p>DVWA also includes a Web Application Firewall (WAF), PHPIDS, which can be enabled at any stage to further increase the difficulty. This will demonstrate how adding another layer of security may block certain malicious actions. Note, there are also various public methods at bypassing these protections (so this can be seen as an extension for more advanced users)!</p>
  <p>There is a help button at the bottom of each page, which allows you to view hints & tips for that vulnerability. There are also additional links for further background reading, which relates to that security issue.</p>
  <hr />
  <br />
</div>
```

Change permissions for DVWA

```
sudo chmod -R 777 /var/www/html/DVWA/
```

Step 3. Install MySQL

The next component for Setting up DVWA is Installing MySQL.

To install MySQL, type the following:

```
sudo apt install mysql-server
```

Note that the installation routine may ask you to create a new password for the root MySQL user. Once you have completed all of the required steps, your MySQL installation should be completed. Let's double-check that our new MySQL server is

running. Type this command:

```
mysql -u root -p
```

Enter the root password you created for MySQL when you installed the software package. Once in, the following to get the server status, version information and more:

```
status
```

This is a good way to ensure that you've installed MySQL and are ready for further configuration.

Restart Apache Server

```
sudo service apache2 restart
```

Create Database and User

To create a MySQL database and user, follow these steps:

At the command line, type the following:

```
mysql -u root -p
```

Type the MySQL root password, and then press Enter.

To create a database, type the following command:

```
CREATE DATABASE dvwadb;
```

To create a database user, type the following command. Replace *dvwausr* with the user you want to create, and replace *dvwa@123* with the

user's password:

```
CREATE USER 'dvwausr'@'127.0.0.1' IDENTIFIED BY 'dvwar@123';
```

Grant permission, type the following command:

Once done, exit the application by typing either of the following commands:

```
\q
```

```
(or)
```

```
exit
```

Step 4. Install PHP5

For our last component in DVWA Installation, we will setup and install PHP. Installing this on your VM is quite easy.

To install PHP, simply type the following command:

```
sudo apt install php5
```

or

```
sudo apt install php5.6
```

Agree to the installation and PHP 5 will be installed on your Server.

Restart Apache Server

```
sudo service apache2 restart
```

Now, let's take a moment to test the PHP software that you just installed. Move into your public web directory:

```
cd /var/www/html
```

Once there, use the text editor to create a file named info.php by typing the following command:

```
sudo vim info.php
```

This command will use the command line editor vim to open a new blank file with this name. Inside this file, type the following:

Inside this file, copy paste the following:

```
<?php phpinfo(); ?>
```

Save your changes by entering:

```
:wq!
```

Once done, open your web browser and type your localhost IP address in the browser.

(Example IP Address)

```
http://127.0.0.1/info.php
```

You will see the default PHP information page, similar to this:



System	Linux DO-Writing 3.16.0-4-amd64 #1 SMP Debian 3.16.7-ckt11-1 (2015-05-24) x86_64
Build Date	Jun 5 2015 11:03:32
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php5/apache2
Loaded Configuration File	/etc/php5/apache2/php.ini
Scan this dir for additional .ini files	/etc/php5/apache2/conf.d
Additional .ini files parsed	/etc/php5/apache2/conf.d/05-opcache.ini, /etc/php5/apache2/conf.d/10-pdo.ini, /etc/php5/apache2/conf.d/20-json.ini, /etc/php5/apache2/conf.d/20-readline.ini
PHP API	20131106
PHP Extension	20131226
Zend Extension	220131226
Zend Extension Build	API220131226,NTS
PHP Extension Build	API20131226,NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	disabled
Zend Memory Manager	enabled
Zend Multibyte Support	provided by mbstring
IPv6 Support	enabled
DTrace Support	enabled
Registered PHP Streams	https, ftps, compress.zlib, compress.bzip2, php, file, glob, data, http, ftp, phar, zip
Registered Stream Socket Transports	tcp, udp, unix, udg, ssl, sslv3, tls, tlsv1.0, tlsv1.1, tlsv1.2
Registered Stream Filters	zlib.*, bzip2.*, convert.iconv.*, string.rot13, string.toupper, string.tolower, string.strip_tags, convert.*, consumed, dechunk

This program makes use of the Zend Scripting Language Engine:
Zend Engine v2.6.0, Copyright (c) 1998-2015 Zend Technologies
with Zend OPcache v7.0.4-dev, Copyright (c) 1999-2015, by Zend Technologies

zend® engine

When you are done looking at this test PHP page, you can remove this file if you want by typing the following command:

```
sudo rm /var/www/html/info.php
```

Install MySQL Extension for PHP.

To Install MySQL Extension for PHP Support, type the following:

```
sudo apt install php5-mysql
```

Once done, you have completed the PHP installation required for DVWA.

Install PHP-GD

DVWA requires a module for PHP which is not installed into Kali Linux or elementaryOS. So we need to add a Debian source for APT.

Once done, you have completed the PHP installation for DVWA.

Step5. Configure DVWA

Now we are ready to edit the source of PHP config files to make sure your web application connects to the database and has got a working captcha. You can obtain reCaptcha keys from your Google Account by [clicking here](#).

We will use the text editor to edit the configuration typing the following command:

```
sudo vim /var/www/html/dvwa/config/config.inc.php.dist
```

Add the database name, user, and password of the MySQL database.

Enter reCaptcha keys.

Here's a screenshot on how your file needs to be after editing.

```
# Thanks to @digininja for the fix.

# Database management system to use
$DBMS = 'MySQL';
#$DBMS = 'PGSQL'; // Currently disabled

# Database variables
# WARNING: The database specified under db_database WILL BE ENTIRELY DELETED during setup.
# Please use a database dedicated to DVWA.
#
# If you are using MariaDB then you cannot use root, you must use create a dedicated DVWA user.
# See README.md for more information on this.
$DVWA = array();
$DVWA[ 'db_server' ] = '127.0.0.1';
$DVWA[ 'db_database' ] = 'dvwa';
$DVWA[ 'db_user' ] = 'dvwausr';
$DVWA[ 'db_password' ] = 'dvwa123';

# Only used with PostgreSQL/PGSQL database selection.
$_DVWA[ 'db_port ' ] = '5432';

# ReCAPTCHA settings
# Used for the 'Insecure CAPTCHA' module
# You'll need to generate your own keys at: https://www.google.com/recaptcha/admin/create
$_DVWA[ 'recaptcha_public_key' ] = '6LeTcWwUAAAAAK4AqeHjdpBIUcM9FKWKHFSBZNbg';
$_DVWA[ 'recaptcha_private_key' ] = '6LeTcWwUAAAAA0gv8Y5BspoV_EJsInw0UIcgRzV';

# Default security level
```

Once done, we need to edit the main config (*php.ini*) file for apache2, which is not correctly overridden for DVWA by default.

```
sudo vim /etc/php5/apache2/php.ini
```

- Enable Allow_url_fopen
- Enable Allow_url_include

This is necessary to exploit the file upload vulnerability. Here's a screenshot for *php.ini* after making changes.

```
; Maximum allowed size for uploaded files.
; http://php.net/upload-max-filesize
upload_max_filesize = 2M

; Maximum number of files that can be uploaded via a single request
max_file_uploads = 20

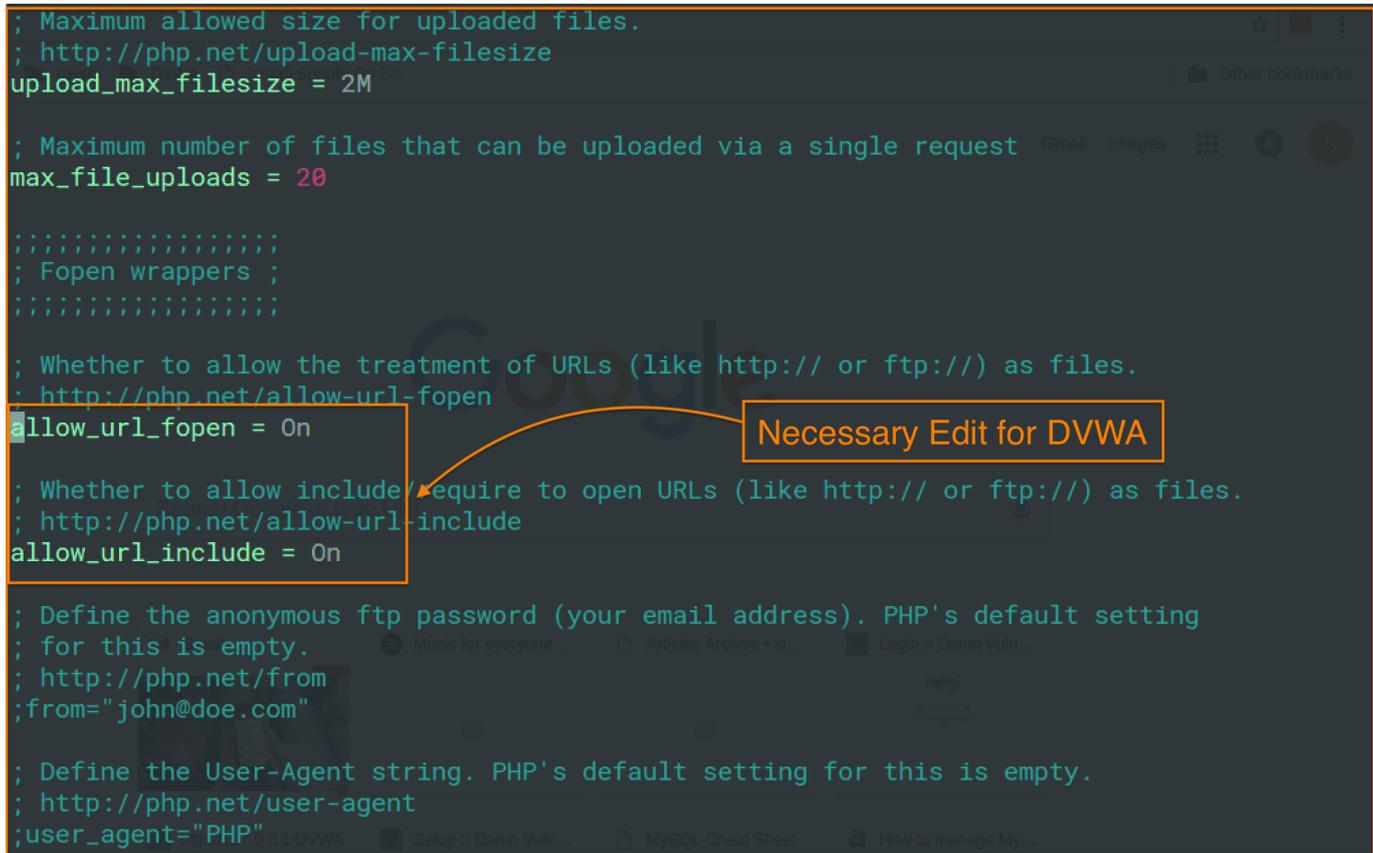
; Fopen wrappers

; Whether to allow the treatment of URLs (like http:// or ftp://) as files.
; http://php.net/allow-url-fopen
allow_url_fopen = On

; Whether to allow include/require to open URLs (like http:// or ftp://) as files.
; http://php.net/allow-url-include
allow_url_include = On

; Define the anonymous ftp password (your email address). PHP's default setting
; for this is empty.
; http://php.net/from
;from="john@doe.com"

; Define the User-Agent string. PHP's default setting for this is empty.
; http://php.net/user-agent
;user_agent="PHP"
```



After saving changes for *php.ini*, we need to follow few more steps.

Install Icweseasel

```
sudo apt install icweseasel
```

Restart Apache

```
sudo /etc/init.d/apache2 restart
```

Restart MySQL Service

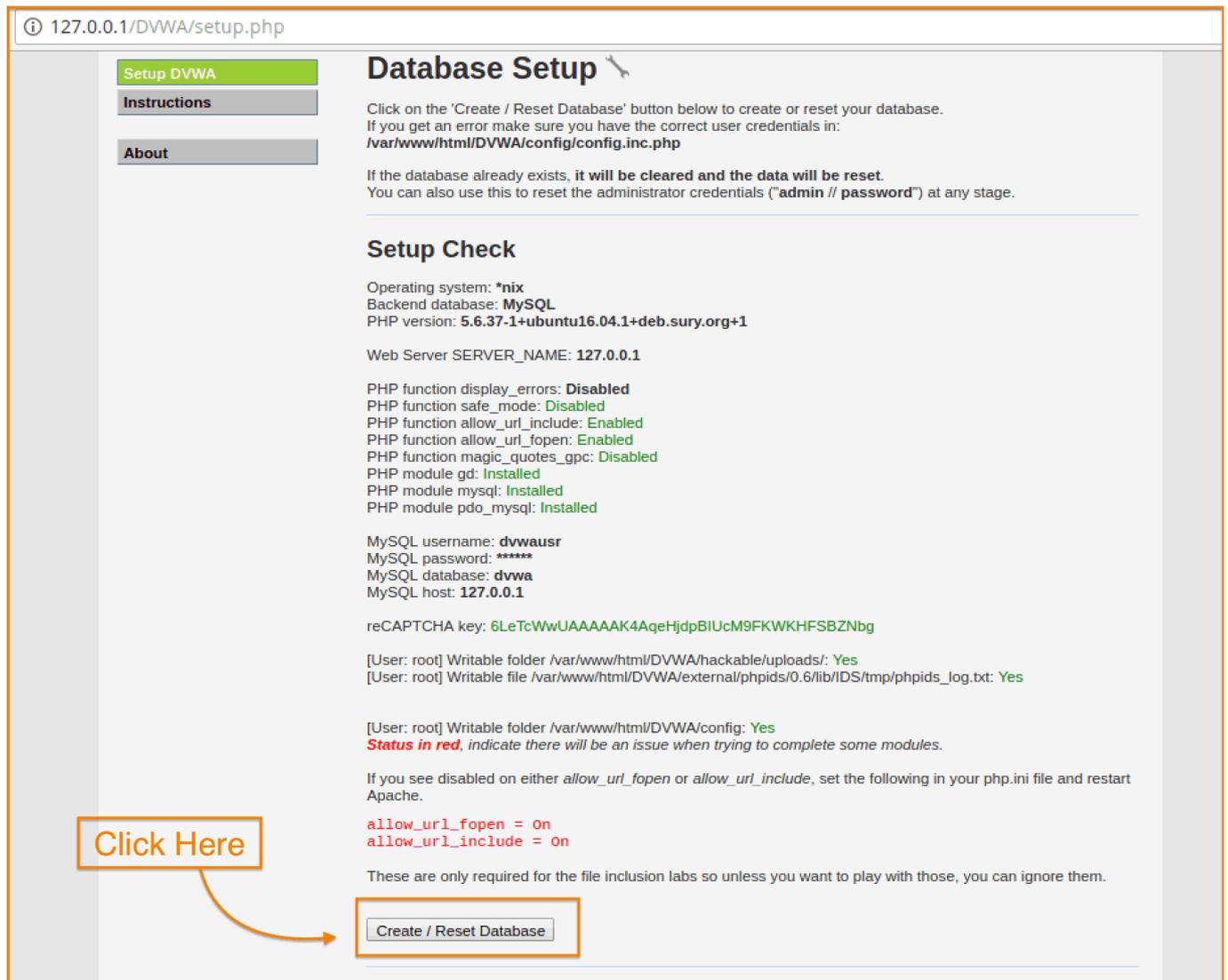
```
sudo /etc/init.d/mysql restart
```

Once done, you have completed the required configuration for DVWA.

Test DVWA Installation

iceweasel http://127.0.0.1/DVWA/setup.php

You will be redirected to the web browser and the page similar to this will be in front of you.



127.0.0.1/DVWA/setup.php

Setup DVWA

Instructions

About

Database Setup

Click on the 'Create / Reset Database' button below to create or reset your database. If you get an error make sure you have the correct user credentials in: `/var/www/html/DVWA/config/config.inc.php`

If the database already exists, **it will be cleared and the data will be reset.** You can also use this to reset the administrator credentials ("**admin // password**") at any stage.

Setup Check

Operating system: ***nix**
Backend database: **MySQL**
PHP version: **5.6.37-1+ubuntu16.04.1+deb.sury.org+1**

Web Server SERVER_NAME: **127.0.0.1**

PHP function display_errors: **Disabled**
PHP function safe_mode: **Disabled**
PHP function allow_url_include: **Enabled**
PHP function allow_url_fopen: **Enabled**
PHP function magic_quotes_gpc: **Disabled**
PHP module gd: **Installed**
PHP module mysql: **Installed**
PHP module pdo_mysql: **Installed**

MySQL username: **dvwausr**
MySQL password: *********
MySQL database: **dvwa**
MySQL host: **127.0.0.1**

reCAPTCHA key: **6LeTcWwUAAAAAK4AqeHjdpBIUcM9FKWKHFSBZNbg**

[User: root] Writable folder /var/www/html/DVWA/hackable/uploads/: **Yes**
[User: root] Writable file /var/www/html/DVWA/external/phpids/0.6/lib/IDS/tmp/phpids_log.txt: **Yes**

[User: root] Writable folder /var/www/html/DVWA/config: **Yes**
Status in red, indicate there will be an issue when trying to complete some modules.

If you see disabled on either `allow_url_fopen` or `allow_url_include`, set the following in your `php.ini` file and restart Apache.

```
allow_url_fopen = on
allow_url_include = on
```

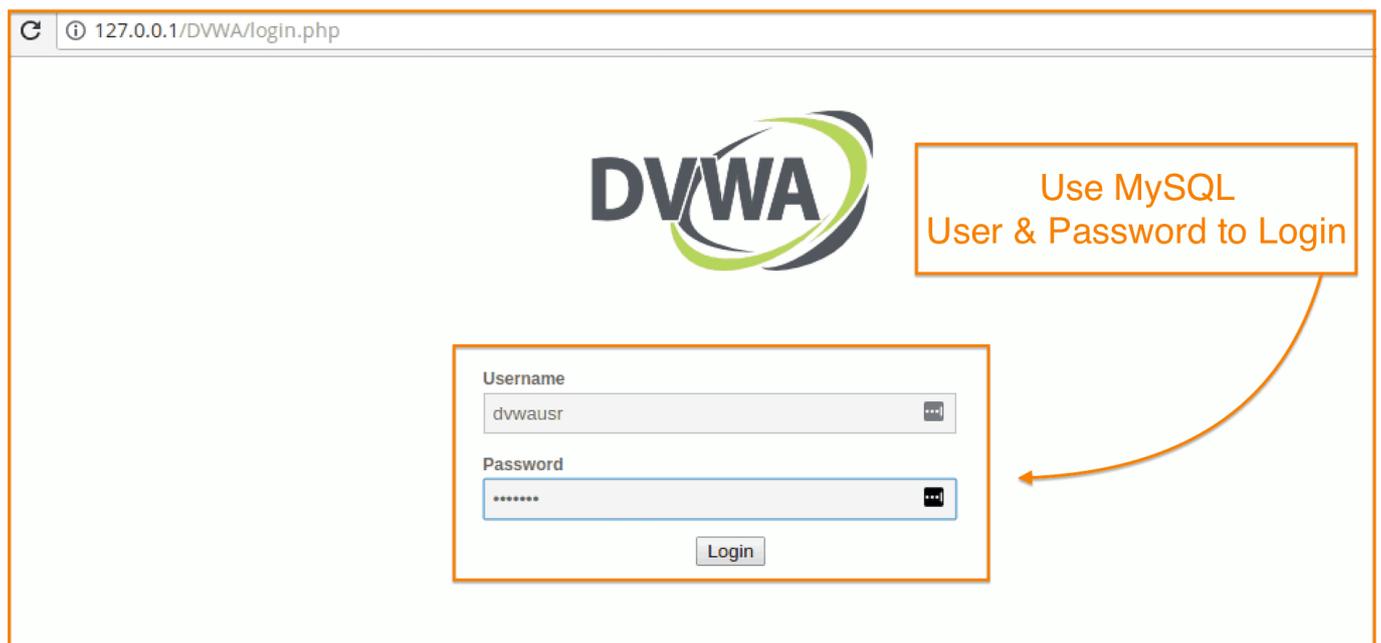
These are only required for the file inclusion labs so unless you want to play with those, you can ignore them.

Click Here

Create / Reset Database

When you are done looking at this DVWA Setup page, you can click on Create / Reset Database button. You will be redirected to the login page.

Use MySQL User and Password to Login



Now, login to change the strength of vulnerabilities by clicking on “DVWA Security”.

DVWA Security Options for Attacks: [Start with Low level.](#)

Low Level: Low-Level Security gives you the freedom to exploit all known vulnerabilities means there will be no security in a given framework and hence you can try all attacks if you are using it first Time.

Medium Level: Medium security will have all entry-level validations and filtration which can stop any script kiddie to get the benefit of available vulnerabilities.

High Level: High Level is kind of Zero Day environment and if you can breach it then that means you are on the right track to becoming a VAPT Expert.

You're done.

So, we have setup a simple vulnerable web application on localhost. You can now *Explore DVWA interface*.

If you encounter any errors during the installation or have questions, Let us know in the comments below!

You may also be interested in reading [How to become a Web Security Researcher](#) or learn [Why questioning is Pivotal to Success in Web Security?](#)

I've also got this [Data Science newsletter](#) that you might be into. I send a tiny email once or twice every quarter with some useful resource I've found.

Don't worry, I hate spam as much as you. Feel free to subscribe.