HackMePlease Walkthrough

Southern Careers Institute - Cyber Security

Adan M. Castaneda  
Jonathan Aceves  
Ngoc-Thao Truong

# Hack Me Please Walkthrough – VirtualBox and VulnHub

An easy machine from Vulnhub and acts like an OSCP, Offensive Security Certified Professional, like machine. <https://www.vulnhub.com/entry/hack-me-please-1,731/>

## 1 Identifying the Target

The target machine was identified through the use of the following command:   
“sudo netdiscover -r 10.0.2.0/24”. The IP addresses provided in the netdiscover command will change based off of the user’s network configurations.

Text

Description automatically generated

Through the discovery of the following IP’s from netdiscover, each IP listed was then searched for in the Mozilla Firefox Browser. It was discovered that the vulnerable machine had the IP address of 10.0.2.5

A screenshot of a computer

Description automatically generated with medium confidence

## 2 Enumerating the Webserver

After confirming the existence and connection of IP 10.0.2.5, a nmap scan was conducted on for the IP address to find any open ports that we may exploit. It was discovered that port 80 HTTP, port 3306 MySQL, and port 33060 MySQL X.

Text

Description automatically generated

Port 3306 being open allows us to exploit this attack vector and attempt to remote connect to the vulnerable machine’s MariaDB server. In order to do this, the source code of the website was analyzed, and files found were opened in hopes of finding clues. One file found in the HTML code named js/main.js seemed to be important. (10.0.2.5/js/main.js)

Graphical user interface, text, application, email

Description automatically generated

Their endpoint is an opensource file management system named seeddms.

Text

Description automatically generated

The following comment line is important to be on the lookout for in the future. Often times, there are numerous bits of information and descriptions of the program being used; this is one of those times.

The walkthrough followed, <https://nepcodex.com/2021/08/hack-me-please-walkthrough-vulnhub/> , stated that the bordered comment line found above is a document management system (DMS) link. If there is a link or description that the user, such as myself, has no knowledge of, it is wise to look up the git repo and for any exploits it may present. In this case, one can search for “seeddms 5.1.22 git repo.”

Git repo: <https://sourceforge.net/p/seeddms/code/ci/5.1.22/tree/>

## 3 Analyzing the Git Repo

Through the Git repo link found above, the directory structure of the framework can be analyzed. One can see the multiple directories where ‘conf’ can be found and this folder contains the configuration of the web app.

Graphical user interface, text

Description automatically generated

The IP address directory tree was then navigated through the use of the commented line and conf. (<http://10.0.2.5/seeddms51x/seeddms-5.1.22/conf>) The IP address would be different based on the user’s network settings in their virtual environment.

Graphical user interface, text, application, email

Description automatically generated

It was found that the requested URL was not located or exists on the web server. This is not an approach that we can exploit, however, being persistent and continuing to search for the conf folder would bring great rewards. By manipulating the web address and going one level back in the directory tree, we find the following server response.

Graphical user interface, text, application, email

Description automatically generated

The forbidden message from the server implies that there is a .htaccess file restricting directory browsing which is found in the repository.

Graphical user interface, text, application, email

Description automatically generated

From the files inside conf folder, we access the .htaccess file and find the following line.

Graphical user interface, text, application

Description automatically generated

One can directly access the path on the target to reveal the actual settings file.

Graphical user interface, text, application

Description automatically generated

The image above shows the database server’s username and password so, we can log into the server. (mysql -h 10.0.2.5 -u<username> -p -D<database\_name>)

Text

Description automatically generated

Using the command, “SHOW TABLES;”, we can look up the database tables found in seeddms.

Text

Description automatically generated

All of the tables listed from the command have a prefix ‘tbl’ but users. We can access and list the records found from “users” with the following command: (SELECT \* FROM users;)

Graphical user interface

Description automatically generated

The password for a user is found but the username associated with the account is not listed. The next step closely follows the walkthrough to gain access into the web app. Since we have database access, we can update the password of the administrator with the command   
(SELECT login,pwd FROM tblUsers;)

Text

Description automatically generated with medium confidence

An online md5 hash generator can be utilized to update the password of admin. <https://www.md5hashgenerator.com/> was used to generate a new md5 hash password. In this case, the same password was followed from the walkthrough, and the md5 hash of “admin” was set as the new password. “21232f297a57a5a743894a0e4a801fc3”

Text

Description automatically generated

After updating the password from the cmd prompt, one can login through the pathway of IP address/seeddms51x/seeddms-5.1.22/

By using the username and password of admin, a user can login to the seeddms database.

Graphical user interface, text, application

Description automatically generated

We are then greeted with a successful login and the file management interface. From the looks of it, we should be able to upload files of our choice which is highly useful for gaining reverse shell access.

Graphical user interface, application

Description automatically generated

## 3 Gaining Root Access with PHP Reverse Shell

To begin with gaining root access, a PHP reverse shell from pentestmonkey was downloaded.   
<https://pentestmonkey.net/tools/web-shells/php-reverse-shell>   
The packaged files were downloaded from the site and set within the downloads folder of the kali linux machine. The files were extracted and the “php-reverse-shell.php” file was modified to fit my machine’s IP address and port I want to listen from.

My machine’s IP address was found using the command “ip a”. The IP we would want to use is the eth0 option. The screenshot shows the eth0 IP address on the left in a kali terminal and the php file in an file window after being modified.

A screenshot of a computer

Description automatically generated with medium confidence

Then, we want to listen on the set port found in the php file. Side note, there are preinstalled reverse shells found in Kali Linux such as “rev.php” which uses port 443 by default.

Graphical user interface

Description automatically generated with medium confidence

The next steps involve uploading the php file into the file management system. Click on the Add document tab found at the very top of the page.

Graphical user interface, website

Description automatically generated

From there, keep all of the settings at their default values and upload the php file into the “Local file” section. Click on the add document, and the file should have been uploaded onto the main page’s content.

\*There has been issues seen where the page does not refresh properly after uploading the document. You can always manually type in the web address again, IP/seeddms51x/seeddms-5.1.22/, or you may click back on the previous arrow. Either way, the file should have been successfully uploaded.\*

Graphical user interface, text, application, email

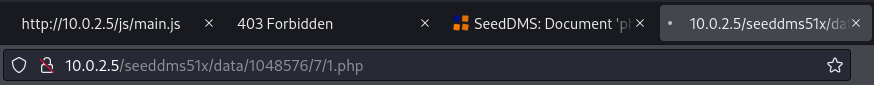
Description automatically generated

Clicking on the file name, the document information can be viewed which is necessary to conduct a reverse shell.

Graphical user interface, application

Description automatically generated

The link required to visit, plug into Mozilla firefox, is <http://IP/seeddms51x/data/1048576/ID/Version.php> Within the link, exchange IP with the IP address of the vulnerable machine, exchange ID with the ID number of the file, and change Version with the Version number associated with the file.



The webpage from the web address provided will continuously load and the kali terminal listening on your designated port will look similar to the following screenshot.

Text

Description automatically generated

We have now successfully gained access with the reverse shell, and we are “www-data.” This is the default user used on Ubunut web servers such as Apache, nginx, etc. for normal operation.

Text

Description automatically generated with medium confidence

With this, we can list out what files are found with the current user by using “ls”. One of the many files that are incredibly useful is the etc file which contains the passwd file. This file can be accessed by using the “grep bash /etc/passwd” command.

\*Make sure to modify the command given your current location in the directory.\*

Text

Description automatically generated

One of the usernames found within the passwd file had been discovered previously through the MariaDB. Thankfully, the password can be easily attained again which is “Saket@#$1337”

With both of the credentials, we can use the command to switch users: “su -l saket” After getting prompted, enter the password and confirm that you have changed user accounts by typing “whoami”.

Text

Description automatically generated

It may come to the surprise of many that the terminal no longer has a “$” before the command line now. This indicates that the terminal is not interactive, and we are currently in a bin/sh. We want to be in a bin/bash environment and to do that we have to conduct extra steps to get around it.

The most common way to spawn a bin/bash environment is to use Python, and we take a look around to see which Python is currently installed on the system.

Graphical user interface, text

Description automatically generated with medium confidence

The terminal states that the system has python2 and python3 installed. Going forward, we use a python2 command which is very common when spawning a bin/bash environment. The command is successful if the current terminal line changes to the following: “saket@ubuntu:~$”

Text

Description automatically generated

The following command “sudo -l” will then tell us what commands we can currently run with the saket user. From the given output, we can run any command as any user in any group.

Text

Description automatically generated

Lastly, we can change to the root user by either using the command “sudo su” or “sudo -i”

Graphical user interface, text

Description automatically generated

We now have root access and have completed the “HackMePlease” vulnerable machine!