Cybersecurity 701

Directory Traversal Lab

Directory Traversal Materials

- Materials needed
  - Kali Linux Virtual Machine
  - Windows 7 Virtual Machine

- Software Tools used
  - DVWA
    - Vulnerable Web Application

Objectives Covered

- Security+ Objectives (SY0-701)
  - Objective 2.4 – Given a scenario, analyze indicators of malicious activity.
    - Application attacks
      - Directory traversal

What is a Directory Traversal Attack?

- When an image is called, it will look like this:
  - filename = images/panda.jpg
    - Could be pulling image from server location: /var/www/html/images/panda.jpg
- What happens if a malicious actor does this?
  - filename=../../penguin.jpg
    - This would be looking for a penguin.jpg image two folders back
- What happens if the malicious person knows the location of a password file?
  - They could gain access to all sorts of ‘secure’ files
  - …if the web server allows directory traversal

Directory Traversal Lab Overview

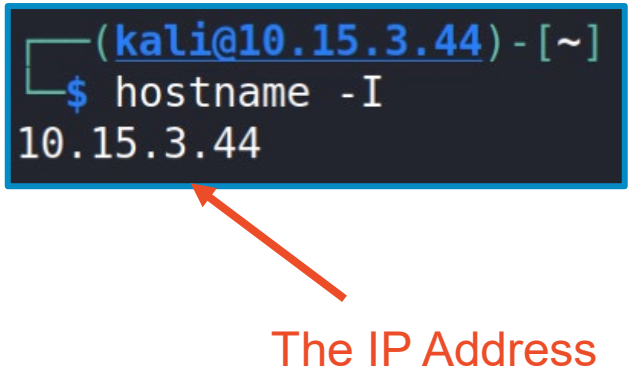
1. Set up Environments
2. Find Linux IP Address
3. Start DVWA Web Services
4. Log in to DVWA & Lower Security
5. Directory Traversal
6. Place Confidential File
7. Directory Traversal (Again)

Set up Environments

- Log into your range
- Open the Kali Linux and Windows 7 Environments
  - You should be on your Kali Linux Desktop
  - You should also be on your Windows 7 Desktop

Locate Linux IP Address

- You will need the IP address of the Kali machine
- Open the Terminal
- In the Linux VM, open the Terminal and type the following command:
  - **`hostname -I`**
- This will display the IP Address
  - Write down the Kali VM IP address



Start the DVWA Web Services

- Start up the web server (on the Kali machine)
  - Use the following command to start XAMPP which will start the services needed to run DVWA

    sudo /opt/lampp/xampp start

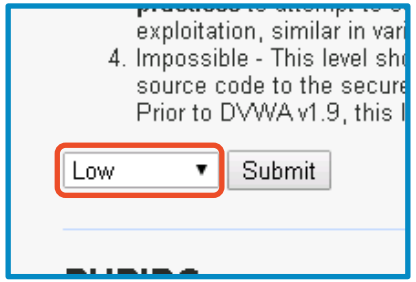
```
┌──(kali@10.15.69.200)-[~]
└─$ sudo /opt/lampp/xampp start
Starting XAMPP for Linux 8.2.4-0...
XAMPP: Starting Apache...ok.
XAMPP: Starting MySQL...ok.
XAMPP: Starting ProFTPD...ok.
```

- On the Windows Machine, go to the DVWA webpage

  http://<Kali-IP-Address>/dvwa

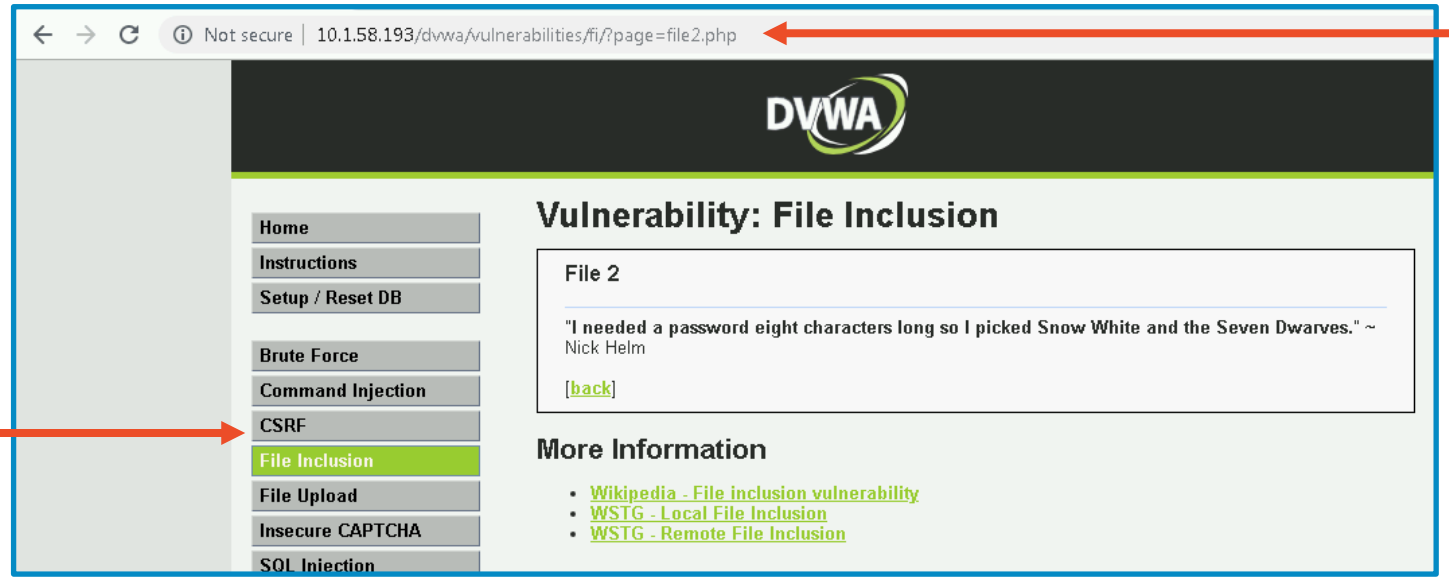
Log in to DVWA & Lower Security

- Login using the following credentials
  - Username: “admin”
  - Password: “password”
- Click on the **DVWA Security** option
- Change the Security Level to **Low**
- Click **Submit**
  - This lowers the DVWA security to the lowest setting



Directory Traversal Setup

- Open a file to view where it is called
  - Click on **File Inclusion**
  - Click on **file2.php**



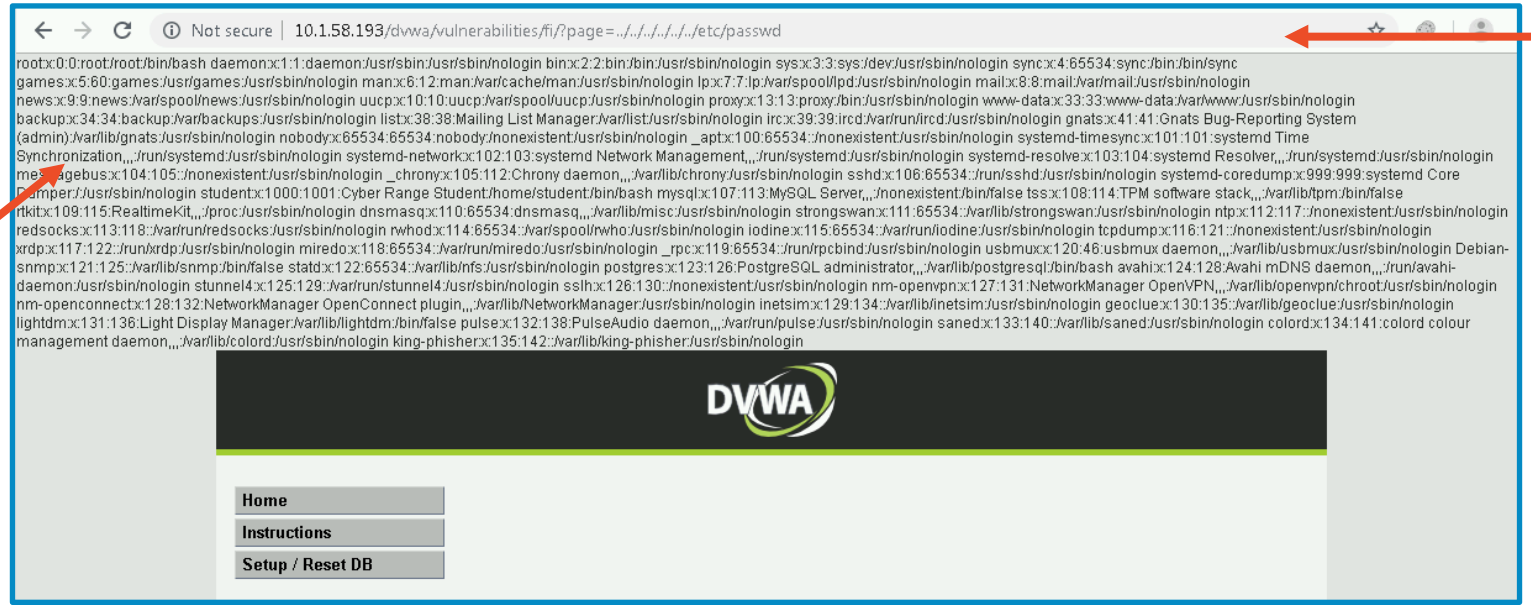
Here is where the website is calling the file

Click on File Inclusion, and then select file2.php

Directory Traversal Example

- Attempt to read the passwd file
  - In the URL, replace **`file2.php`** with the following:

    `../../../../../../etc/passwd`



This is an example of directory traversal

Notice the passwd file is being displayed

Another Directory Traversal Example

- Attempt to figure out the version of the system
  - In the URL, replace the traversal after “page=“ with the following:

    `../../../../../../proc/version`

Create Confidential File

- Go to the Kali Machine
- Open a Terminal
- Navigate to the Desktop

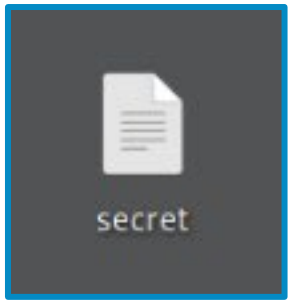
  `cd Desktop`

- Place a secret file

  `touch secret`

```
┌──(kali@10.15.92.122)-[~]
└─$ cd Desktop

┌──(kali@10.15.92.122)-[~/Desktop]
└─$ touch secret
```



Verify a file named “secret” appears on the Desktop

Edit Confidential File

- Edit the file
  `nano secret`
- Write fake credentials
- Save and Exit
  `CTRL+X`
  `Y`
  `ENTER`

```
┌──(kali@10.15.92.122)-[~/Desktop]
└─$ nano secret
```

Fake credentials →

```
  GNU nano 7.2
US Bank Login Info
Username: MLowery19
Password: UncMike4L
```

Directory Traversal to Read Secret File

- Go to the Windows system
- Attempt to read the secret file

  `../../../../../../home/USERNAME/Desktop/secret`

  - The **USERNAME** is likely **kali** for hosts on the CYBER.ORG range

How to Defend Against a Directory Traversal Attack?

- Validate inputs!
  - Don’t allow directory traversal on the web server
  - Don’t allow any “../../”
- Validate the base directory
  - All files should come from a certain location
  - Verify the location of the file
- What are some other ways of defending against directory traversal?