Cybersecurity 701

XSS DVWA Lab

CYBERSECURITY & INFRASTRUCTURE SECURITY AGENCY
CISA

CYBER.ORG
THE ACADEMIC INITIATIVE OF THE CYBER INNOVATION CENTER

XSS DVWA Materials

- Materials needed
  - Kali Virtual Machine (with DVWA)
  - Windows 7 Virtual Machine

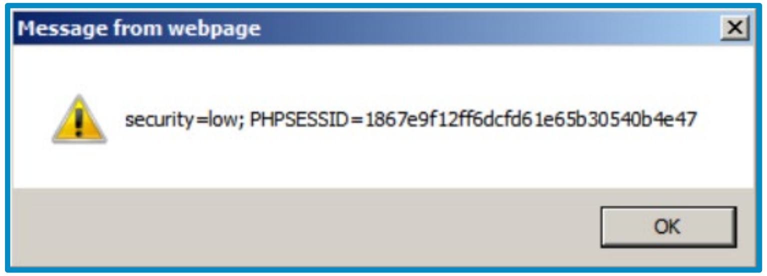
- Software Tool used
  - DVWA (Damn Vulnerable Web Application)
    - Follow the DVWA Setup Lab if not previously installed/available on your VM

Objectives Covered

- Security+ Objectives (SY0-701)
  - Objective 2.3 - Explain various types of vulnerabilities.
    - Web-based
      - Cross-site scripting (XSS)

What is a Cross-Site Scripting Attack?

- Inserting scripts (usually JavaScript) into a pages' HTML to bypass server access controls
- Can be used to access data that should be hidden on a webpage
  - Why is this dangerous if the user is privileged?



security=low; PHPSESSID=1867e9f12ff6dcfd61e65b30540b4e47

Cross-Site Scripting Lab Overview

1. Set up environments
2. Access DVWA website
3. Lower DVWA security
4. XSS (Reflected)
5. XSS (Reflected) with HTML
6. XSS (Reflected) Vulnerability

Set up Environments

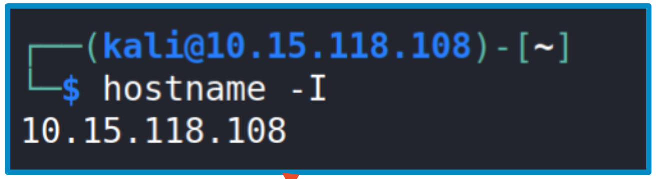
- Log into your range
- Open the Kali Linux and Windows 7 Environments
  - You should be on your Kali Linux Desktop
  - You should also be on your Windows 7 Desktop

Find the IP Address (Kali Machine)

- 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
  ```

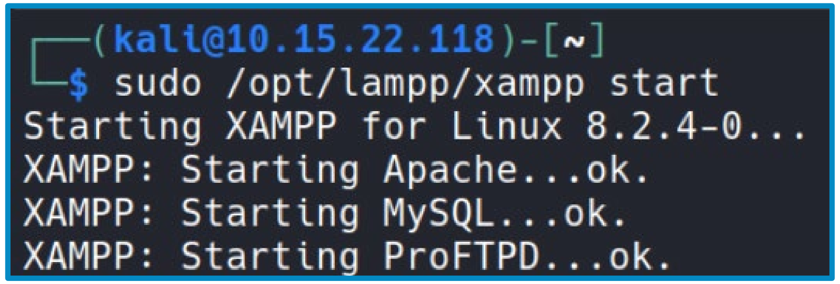
- This will display the IP Address
  - Write down the Kali VM IP address



```
┌──(kali@10.15.118.108)-[~]
└─$ hostname -I
10.15.118.108
```

The IP Address

Log into DVWA



```
┌──(kali@10.15.22.118)-[~]
└─$ 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.
```

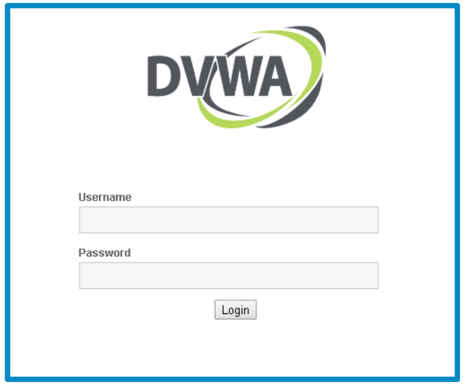
- Start up the web servers (on the Kali machine)
  - Use the following command to start XAMPP:

    `sudo /opt/lampp/xampp start`

- On the Windows Machine, use Google Chrome to go to the DVWA webpage

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

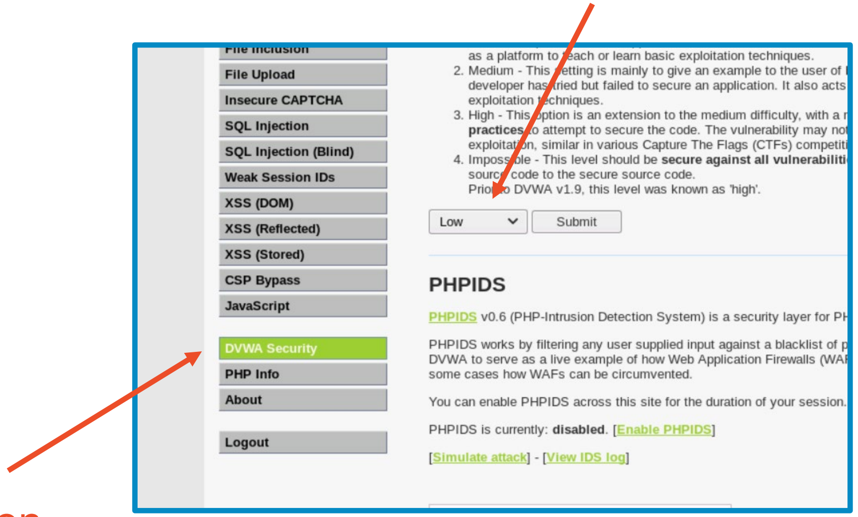
- Login credentials are **admin/password**



Lower DVWA's Security

- Click on the *DVWA Security* button
- Change the security drop down option to *Low*
- Select *Submit* button to set the website vulnerability

Set to Low



as a platform to teach or learn basic exploitation techniques.
2. Medium - This setting is mainly to give an example to the user of I developer has tried but failed to secure an application. It also acts exploitation techniques.
3. High - This option is an extension to the medium difficulty, with a r **practices** to attempt to secure the code. The vulnerability may not exploitation, similar in various Capture The Flags (CTFs) competiti
4. Impossible - This level should be **secure against all vulnerabiliti** source code to the secure source code. Prior to DVWA v1.9, this level was known as 'high'.

Low ⌄ Submit

PHPIDS

PHPIDS v0.6 (PHP-Intrusion Detection System) is a security layer for PH

PHPIDS works by filtering any user supplied input against a blacklist of p DVWA to serve as a live example of how Web Application Firewalls (WAF some cases how WAFs can be circumvented.

You can enable PHPIDS across this site for the duration of your session.

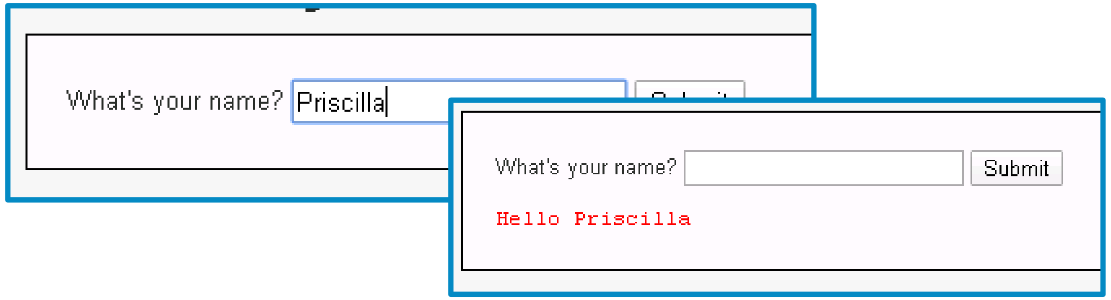
PHPIDS is currently: **disabled**. [Enable PHPIDS]

[Simulate attack] - [View IDS log]

DVWA Security button

XSS (Reflected)

- Click on the XSS (Reflected) Tab
- You should see a “What’s your name?” prompt
  - Enter your name
    - You should see “Hello Your-Name” appear
    - This is how the prompt is supposed to work



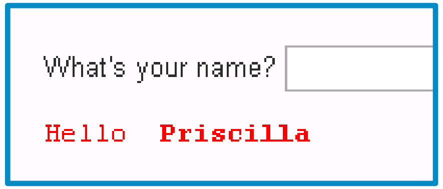
What's your name? Priscilla

What's your name?

Hello Priscilla

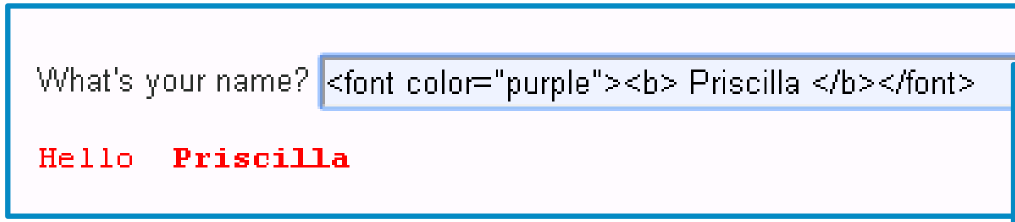
XSS (Reflected) with HTML

- Now, mess with the HTML, make the font bold
  - Search for `<b> Your-Name </b>`
    - You should see your name in bold font now
- What if we want to change the color of the font (keep bold too)?
  - Search for `<font color="purple"><b> Your-Name </b></font>`



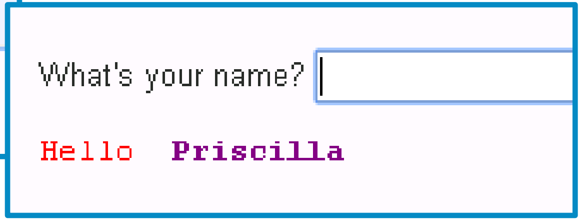
What's your name?

Hello **Priscilla**



What's your name? <font color="purple"><b> Priscilla </b></font>

Hello **Priscilla**

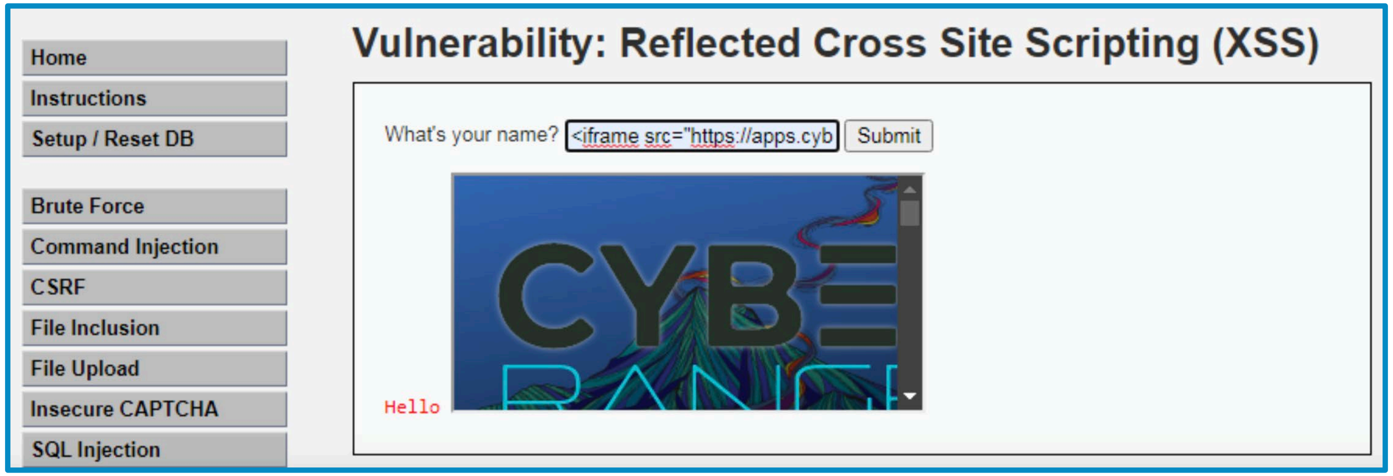


What's your name?

Hello **Priscilla**

XSS (Reflected) with HTML

- Now, display a website
  - Search for `<iframe src="website-URL"></iframe>`
  - You should see a website load in the frame
    - Note it may take a moment for the website to load



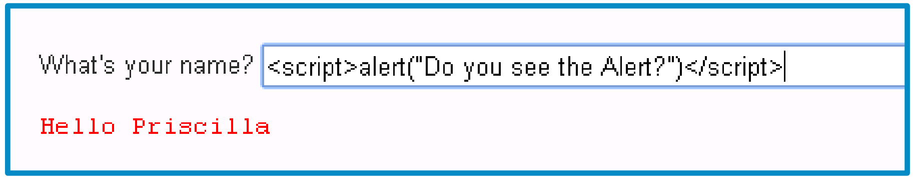
Vulnerability: Reflected Cross Site Scripting (XSS)

What's your name? <iframe src="https://apps.cyb

Hello

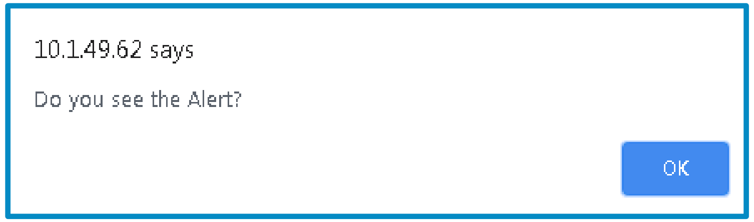
XSS (Reflected) with HTML

- What if we want to display a page alert?
  - Search `<script>alert("Your-Message")</script>`
    - You should for see an alert appear with your message in the box



What's your name? <script>alert("Do you see the Alert?")</script>

Hello Priscilla

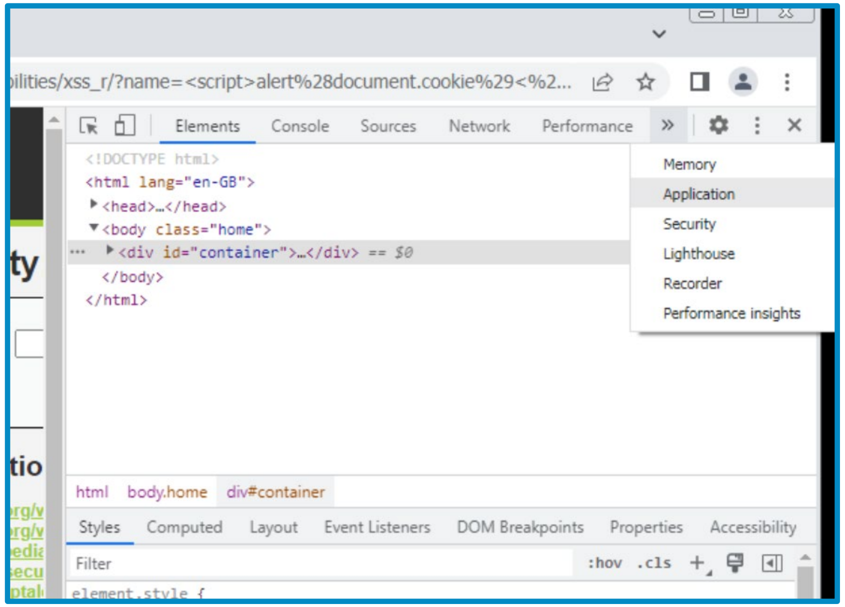


10.1.49.62 says

Do you see the Alert?

XSS (Reflected) Vulnerability - Setup

- Right click on the page and select “Inspect” to open the Developer’s tools
- Click on the “Applications” tab which may require clicking the double arrow to open the dropdown options



ilities/xss_r/?name=<script>alert%28document.cookie%29<%2...

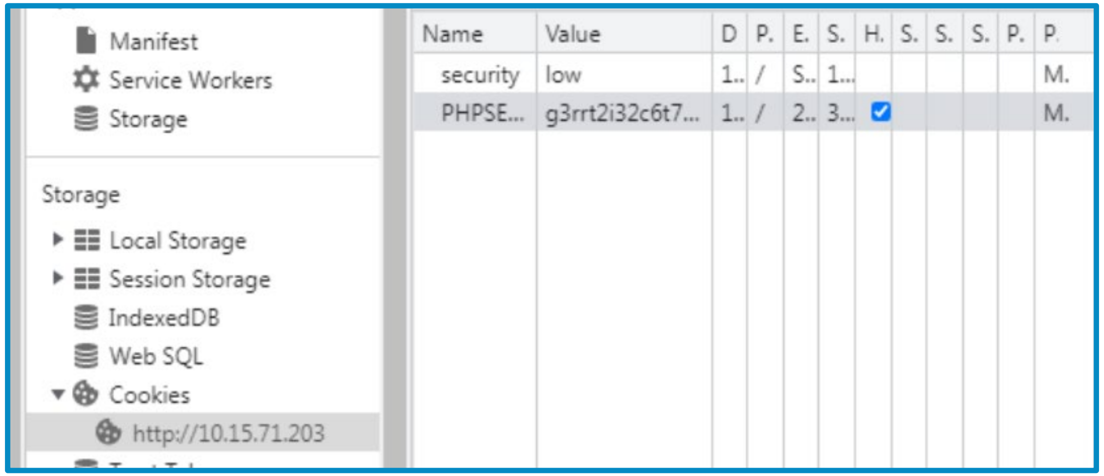
```
<!DOCTYPE html>
<html lang="en-GB">
 ▶<head>…</head>
 ▼<body class="home">
…  ▶<div id="container">…</div> == $0
  </body>
</html>
```

html body.home div#container

```
element.style {
```

XSS (Reflected) Vulnerability – Setup cont’d

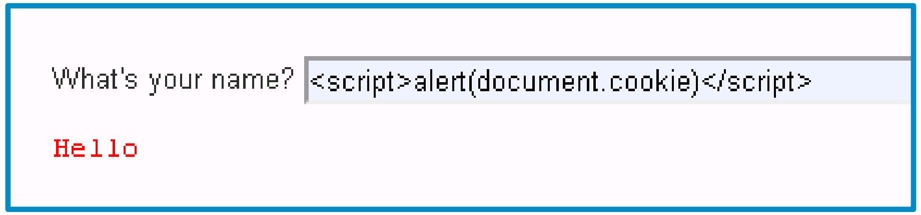
- Click on the arrow next to Cookies and select the link for DVWA, then select the PHPSESSID row
- Click on the 5th column, HttpOnly, and uncheck the box by double clicking or hitting the spacebar
- Close the Developer’s tools window



Name	Value	D	P.	E.	S.	H.	S.	S.	S.	P.	P.
security	low	1..	/	S..	1...						M.
PHPSE...	g3rrt2i32c6t7...	1..	/	2..	3...	☑					M.

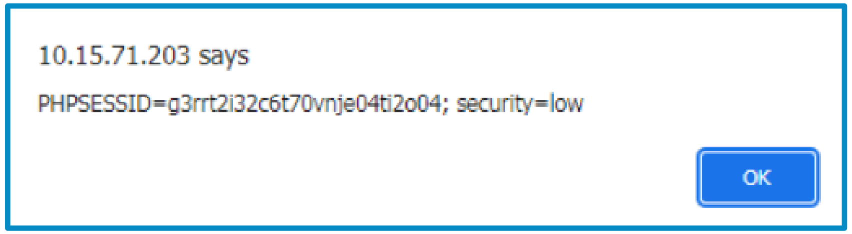
XSS (Reflected) Vulnerability

- What is a vulnerability here?
  - An attacker can grab the Session ID
- Obtain the Session ID
  - Search for `<script>alert(document.cookie)</script>`
    - This displays the cookies for the website



What's your name? <script>alert(document.cookie)</script>

Hello

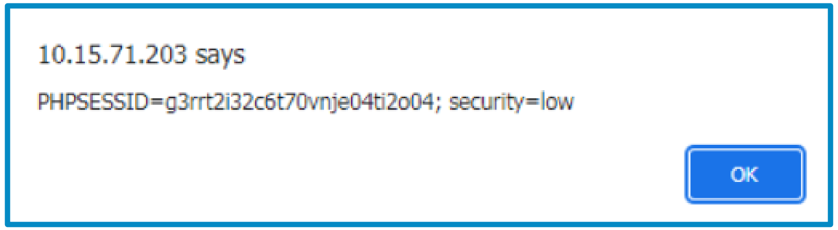


10.15.71.203 says

PHPSESSID=g3rrt2i32c6t70vnje04ti2o04; security=low

XSS (Reflected) Vulnerability

- Why is obtaining the session ID bad?
  - What can a hacker do with this information?
- The session ID is created when a user logs into the website.
  - Server says: "The user is authenticated. Here's a cookie so they don't have to login again and again. I'll use the session ID to remember who they are."
- The session ID can be used to bypass the log-in!
  - What happens if a hacker obtains a session ID for someone logged into a bank's website or some other security-sensitive organization?



10.15.71.203 says

```
PHPSESSID=g3rrt2i32c6t70vnje04ti2o04; security=low
```

Defending Against a Cross-Site Scripting Attack

- Sanitize the inputs!
  - Reject inputs that are not what the search was meant for
  - NEVER trust user input – check it
  - “Escape” the user’s input
    - Escaping the user’s input will not run the data as HTML
    - Will not interpret the HTML
    - This will just display whatever was typed as is
    - Will display characters as they are
      - What is the importance of these characters: “<“ and “>” in a XSS attack?
- What are some other ways of defending against a Cross-Site Scripting attack?