# Report on File Inclusion Vulnerabilities Using DVWA

## Introduction

In this exercise, I explored Local File Inclusion (LFI) and Remote File Inclusion (RFI) vulnerabilities on the DVWA platform to understand their impact and mechanics.

## Task 1: Local File Inclusion (LFI)

By manipulating the URL parameter to access /etc/passwd, I was able to retrieve the contents of the system’s password file. The output showed a list of system users such as:

root:x:0:0:root:/root:/usr/bin/zsh  
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin  
bin:x:2:2:bin:/bin:/usr/sbin/nologin  
sys:x:3:3:sys:/dev:/usr/sbin/nologin  
...  
kali:x:1000:1000:,,,:/home/kali:/usr/bin/zsh

This confirmed the presence of an LFI vulnerability because the web app included system files based on user input without proper validation or sanitization. It demonstrated how an attacker could expose sensitive system information through simple URL manipulation.

## Task 2: Remote File Inclusion (RFI)

I set up a PHP reverse shell (shell.php) on my local machine, listening on port 8000. Injecting the URL of this shell into the vulnerable parameter on DVWA allowed me to execute system commands remotely on the server. For example, running the command id returned:

uid=1000(kali) gid=1000(kali) groups=1000(kali),4(adm),27(sudo),...

This showed me the user context under which the web server process was running, confirming that RFI enabled remote command execution. Some initial PHP warnings appeared due to missing command parameters, but once fixed, the shell operated correctly.

## Lessons Learned

- The exact listing of system users highlights how file inclusion can leak sensitive system-level data.  
- RFI can be exploited to run arbitrary commands remotely, leading to potential system takeover.  
- Proper input validation and server configuration (e.g., disabling allow\_url\_include) are essential defenses.  
- Real-world error messages and warnings provide attackers with clues and should be minimized in production environments.

## Conclusion

This hands-on experience deepened my understanding of how LFI and RFI work, the risks they pose, and the importance of securing web applications through validation and configuration. It also gave me practical insight into attacker methods, making me better equipped to recognize and mitigate such vulnerabilities.