**Verifica della Vulnerabilità di File Upload su DVWA tramite Shell PHP**

**Obiettivo**: Verificare la vulnerabilità di **File Upload** su **Damn Vulnerable Web Application (DVWA)** utilizzando una shell PHP per ottenere un controllo remoto su **Metasploitable** da **Kali Linux**.

**Passaggi Eseguiti**

**1. Verifica della Comunicazione tra Kali Linux e Metasploitable**

• **Scopo**: Assicurarsi che le macchine virtuali Kali Linux e Metasploitable siano in grado di comunicare tra loro.

• **Procedura**: Dal terminale di Kali Linux, è stato eseguito un comando ping verso l’indirizzo IP della macchina Metasploitable.

• **Comando**: ping 192.168.1.11

• **Procedura**: Utilizzando il browser su Kali Linux, è stato inserito l’indirizzo IP di Metasploitable per accedere a DVWA (<http://192.168.1.11/dvwa>).

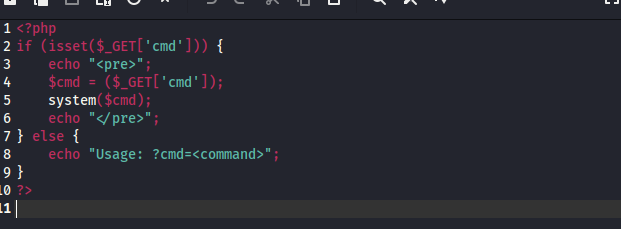
• **Configurazione**: Una volta effettuato l’accesso a DVWA, è stato impostato il **livello di sicurezza su “Low”** per abilitare il caricamento di file senza particolari restrizioni.

**2. Creazione della Shell PHP**

• **Scopo**: Creare una shell PHP che, una volta caricata su DVWA, possa essere utilizzata per eseguire comandi sul sistema Metasploitable.

• **Procedura**: È stato creato un file PHP denominato shell.php con il codice come da immagine

• **Funzionalità del Codice**: Questo codice consente l’esecuzione di comandi remoti attraverso il parametro cmd passato nell’URL, permettendo il controllo remoto del sistema su cui il file è caricato.

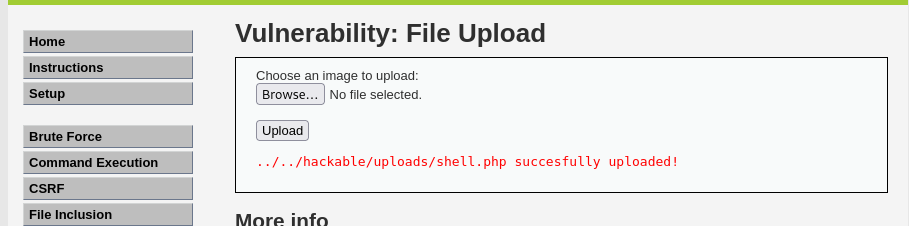


**3. Caricamento della Shell PHP su DVWA**

• **Scopo**: Verificare la possibilità di caricare il file shell.php sulla piattaforma DVWA tramite il modulo di File Upload, sfruttando la vulnerabilità di upload.

• **Procedura**: Il file shell.php è stato selezionato e caricato tramite la funzionalità di upload presente nella sezione **File Upload** di DVWA.

• **Risultato**: Il file shell.php è stato caricato con successo, come confermato dal messaggio di successo visualizzato nell’interfaccia di DVWA.

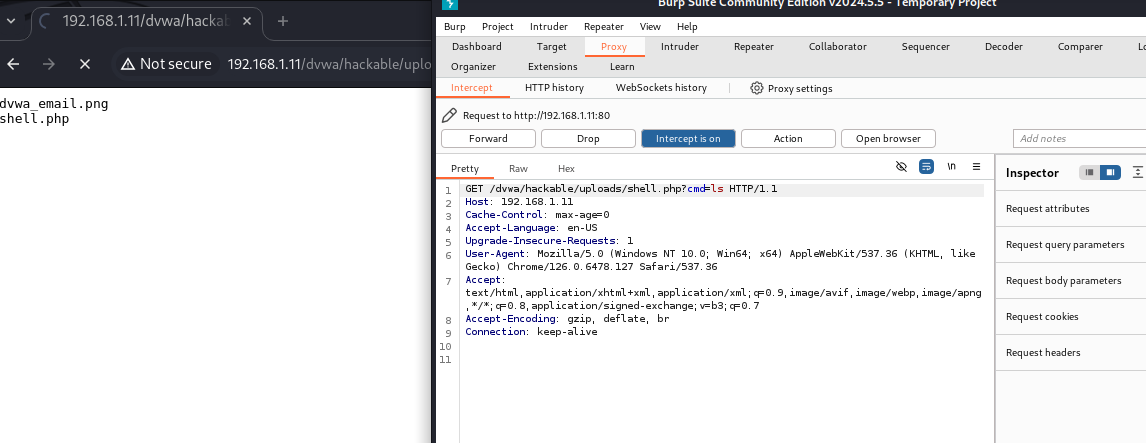


**4. Test della Shell PHP tramite Burp Suite**

• **Scopo**: Utilizzare **Burp Suite** per intercettare e modificare le richieste HTTP alla shell PHP.

• **Azione**: Con Burp Suite in modalità **Intercept**, è stato inserito nel browser l’URL del file caricato, aggiungendo ?cmd=ls alla fine dell’URL per eseguire il comando ls: 192.168.1.11/dvwa/hackable/uploads/shell.php?cmd=ls

• **Risultato**: La richiesta è stata intercettata e inoltrata tramite Burp Suite, e l’output del comando ls è stato visualizzato correttamente sulla pagina web, come mostrato nello screenshot.

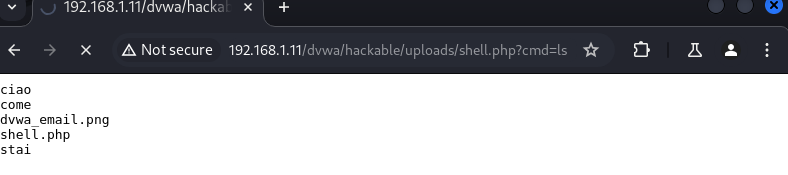


**5. Esecuzione di Comandi Personalizzati**

• **Scopo**: Verificare la possibilità di eseguire ulteriori comandi tramite la shell PHP caricata.

• **Azione**: Nella barra degli indirizzi, è stato inserito il seguente comando per creare un file vuoto con il nome “ciao come stai”:

• **Risultato**: Inoltrando la richiesta con Burp Suite, il comando touch ciao come stai è stato eseguito con successo, creando il file “ciao come stai” sul server. Il risultato è stato visibile direttamente attraverso l’interfaccia di DVWA.



**Conclusioni**

Il test ha dimostrato che la vulnerabilità di **File Upload** su DVWA, quando il livello di sicurezza è impostato su “Low”, consente l’inserimento di una shell PHP con cui eseguire comandi remoti sul server Metasploitable. Questa vulnerabilità permette a un attaccante di ottenere potenzialmente il controllo remoto del sistema, compromettendo la sicurezza del server. L’utilizzo di **Burp Suite** è stato essenziale per intercettare e modificare le richieste HTTP, confermando l’efficacia dell’exploit.

Questi risultati evidenziano l’importanza di adeguate misure di sicurezza nei sistemi di upload e il ruolo critico della configurazione del livello di sicurezza per prevenire accessi non autorizzati.