

Burp Suite

Target y Proxy

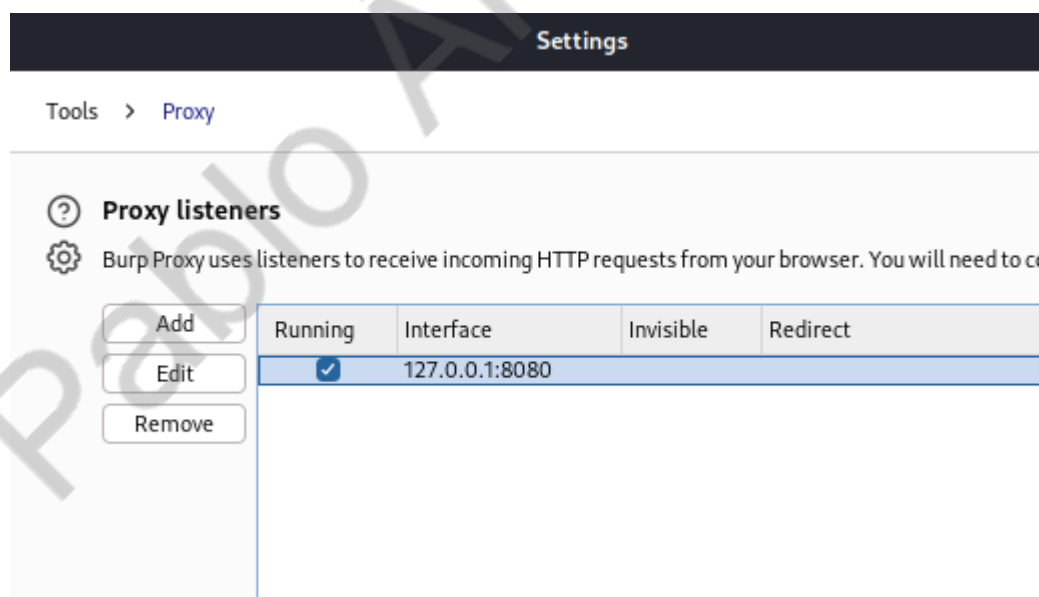
Paso 1:

```
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:d3:83:07
          inet addr:10.0.2.7  Bcast:10.0.2.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fed3:8307/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:35 errors:0 dropped:0 overruns:0 frame:0
          TX packets:65 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:4629 (4.5 KB)  TX bytes:6823 (6.6 KB)
          Base address:0xd020 Memory:f0200000-f0220000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:91 errors:0 dropped:0 overruns:0 frame:0
          TX packets:91 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:19301 (18.8 KB)  TX bytes:19301 (18.8 KB)
```

Lo primero que haremos será iniciar la máquina de Metasploitable 2 y en honor al tiempo revisar su dirección IP. Siendo la 10.0.2.7

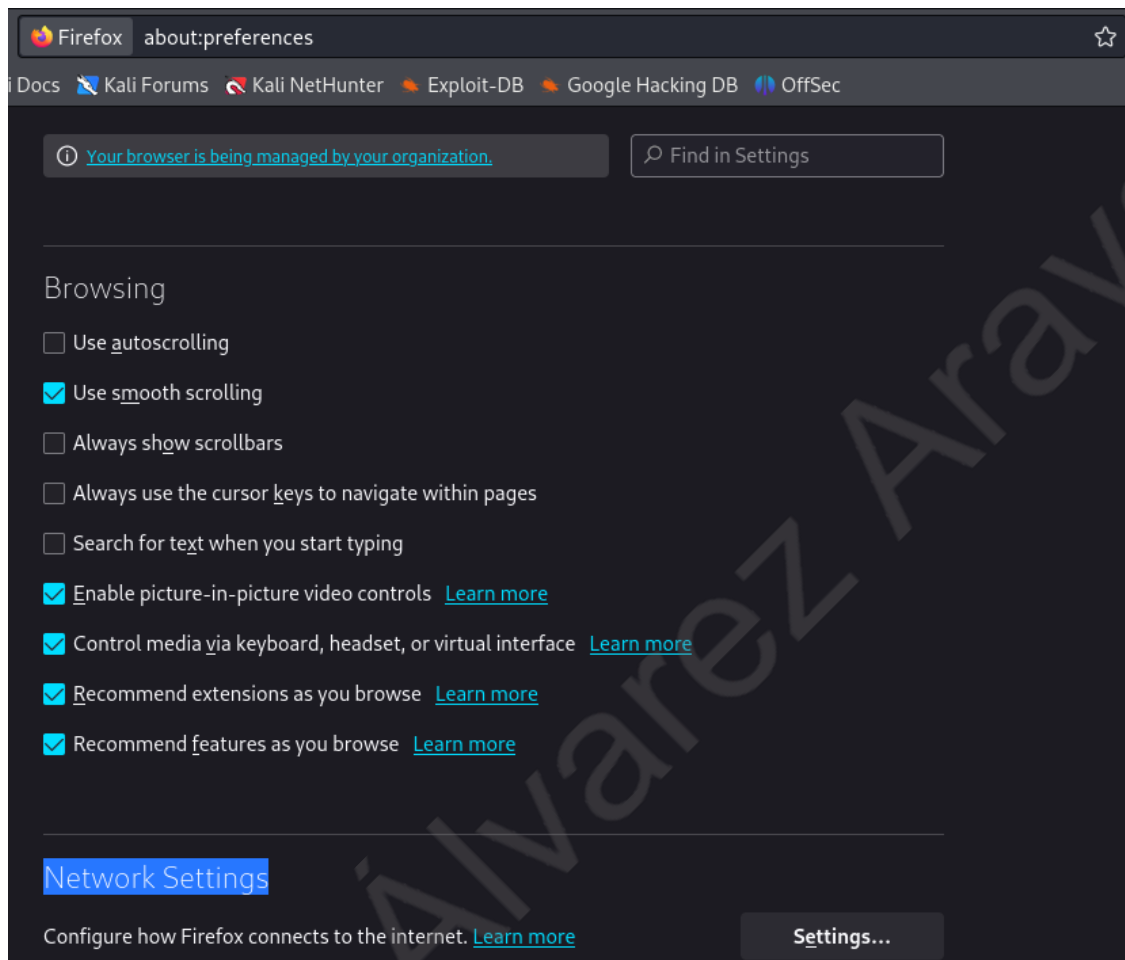
Paso 2:



Vamos a kali linux, abrimos Burp Suite y nos dirigimos a proxy y después a settings donde podemos ver que la dirección IP que se usa por defecto en Burp Suite es la de localhost.

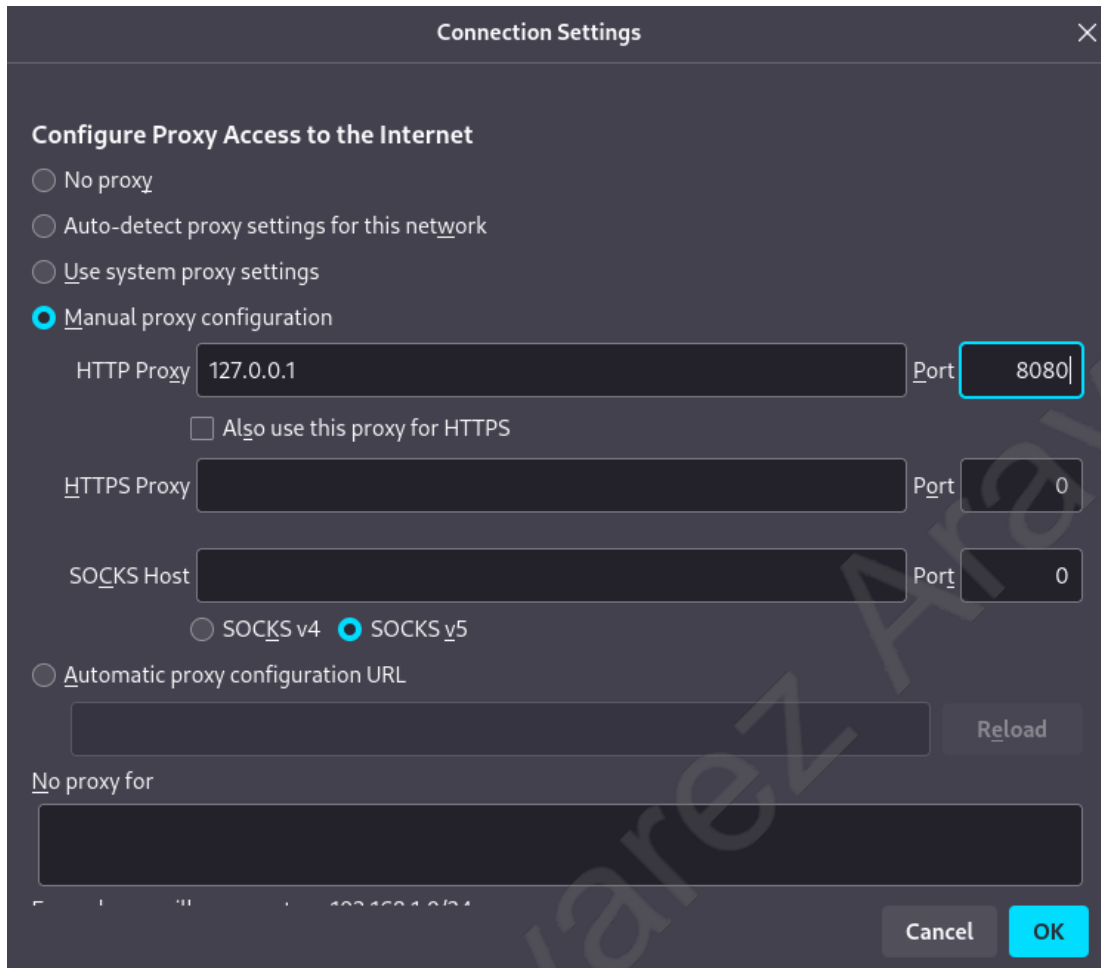
Paso 3:

Y lo siguiente que haremos será configurar nuestro navegador Firefox para que primero tenga que pasar por ese proxy, es decir, esa dirección IP.



Para eso nos dirigimos a firefox y nos vamos a settings, luego bajamos hasta donde dice Network Settings y volvemos a pulsar en settings...

Paso 4:



Habilitamos la opción de Manual proxy configuration, especificamos la IP y el puerto y presionamos en OK.

Y de esta manera ya tenemos configurado el navegador para que pase por el proxy Burp Suite.

Paso 5:

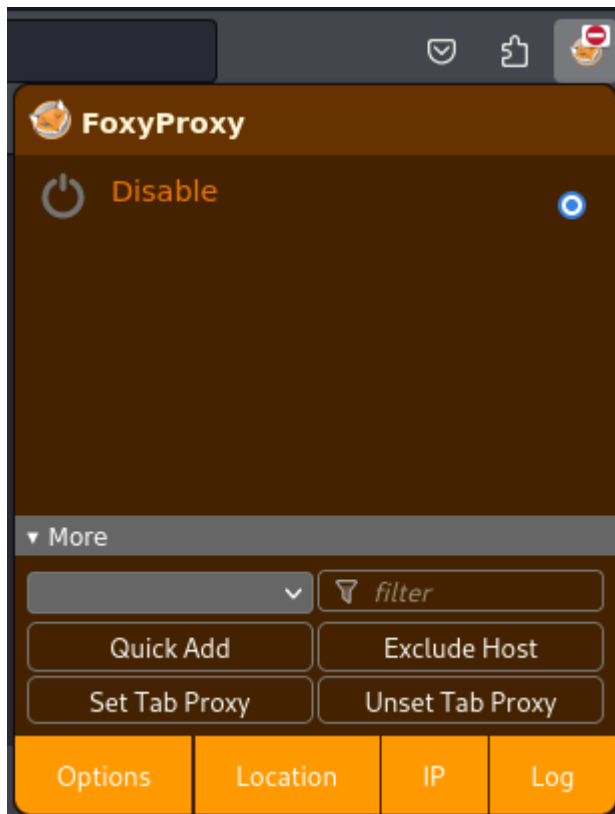
Cerramos el navegador y lo volvemos a abrir para que se efectúen los cambios.

Instalar Foxy Proxy

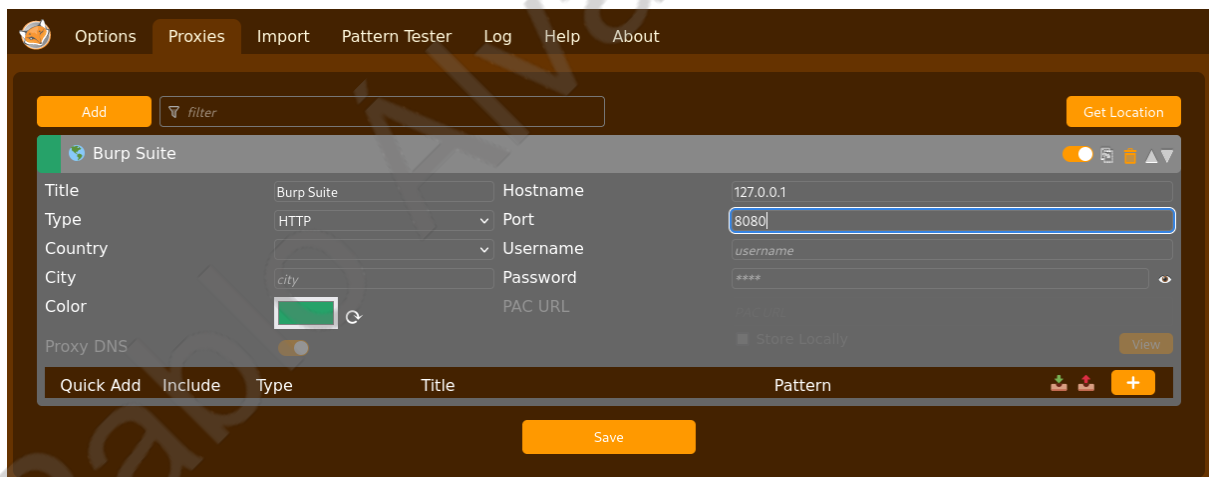
Paso 1:

Agregamos la extensión

<https://addons.mozilla.org/es/firefox/addon/foxyproxy-standard/>



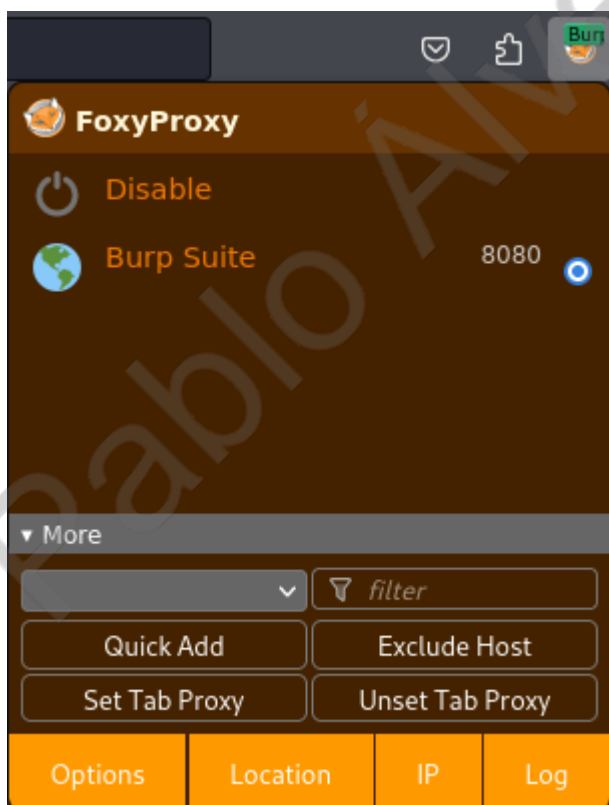
Y una vez que lo hagamos lucirá como se aprecia en la imagen, luego presionamos en Options.



Dentro de las opciones nos vamos a Proxies, presionamos en Add, completamos el nombre, la dirección, el puerto y le damos a Save.



Cerramos la pestaña y si nos dirigimos a otra vemos que se ha agregado correctamente el proxy de Burp Suite.

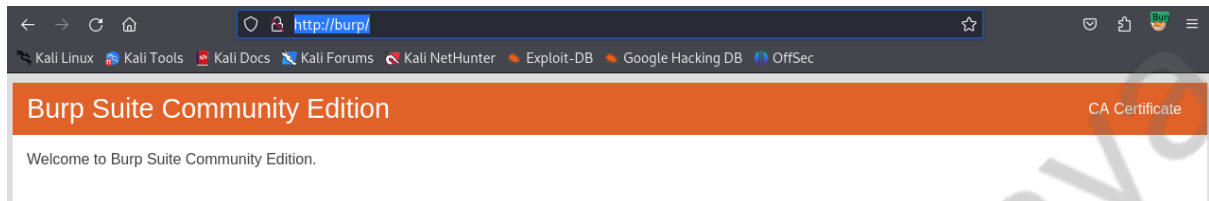


Si lo seleccionamos vemos que ya se encuentra habilitado.

Instalar CA Certificate

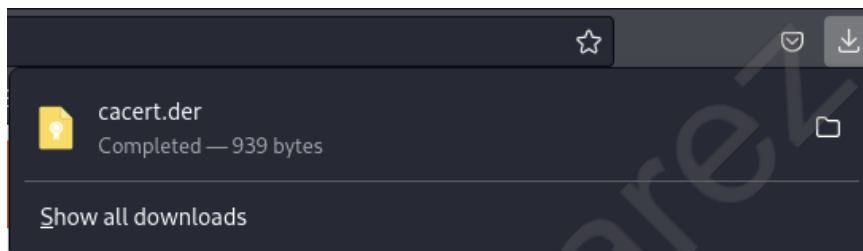
Paso 1:

Para acceder a páginas con protocolo HTTPS es necesario instalar un certificado, para eso nos dirigimos a <http://burp/>



Paso 2:

Pulsaremos en donde dice CA Certificate y se nos descargará un certificado.

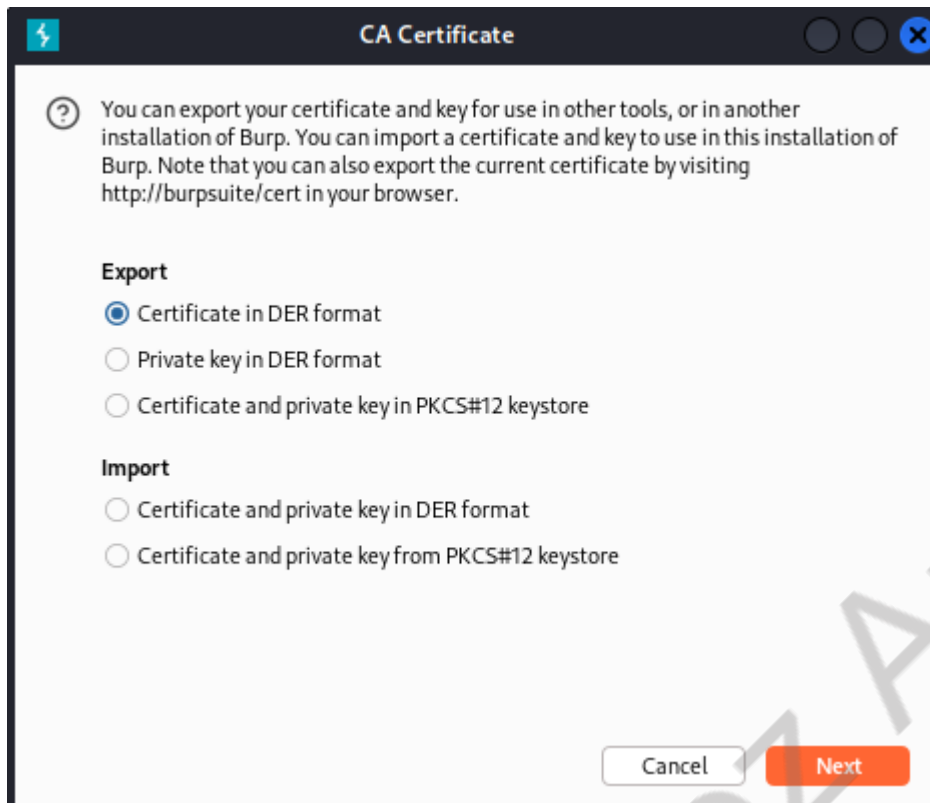


Paso 3:



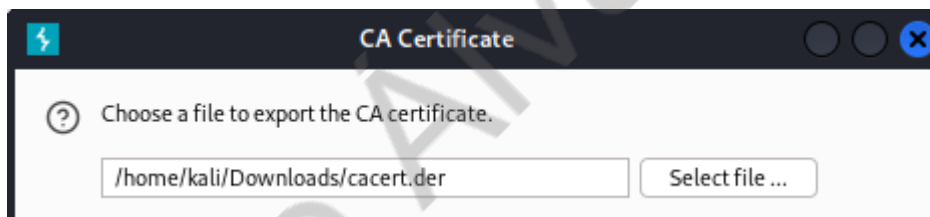
Volvemos a Burp Suite y en las configuraciones del proxy pulsamos en Import/ export CA certificate

Paso 4:



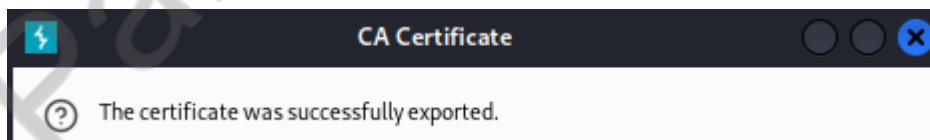
Seleccionamos la primera opción de Certificate in DER format y le damos en Next.

Paso 5:



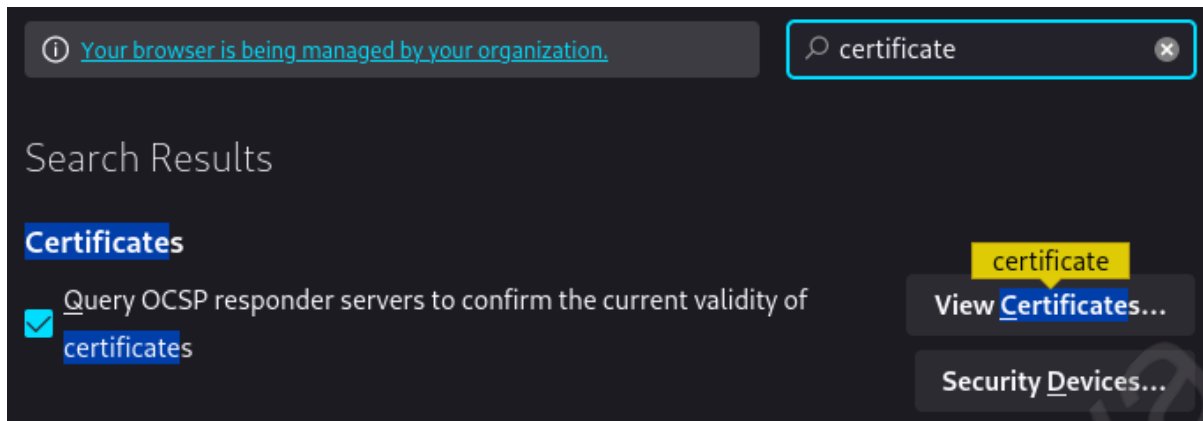
Seleccionamos el certificado que descargamos y le damos en Next.

Paso 6:

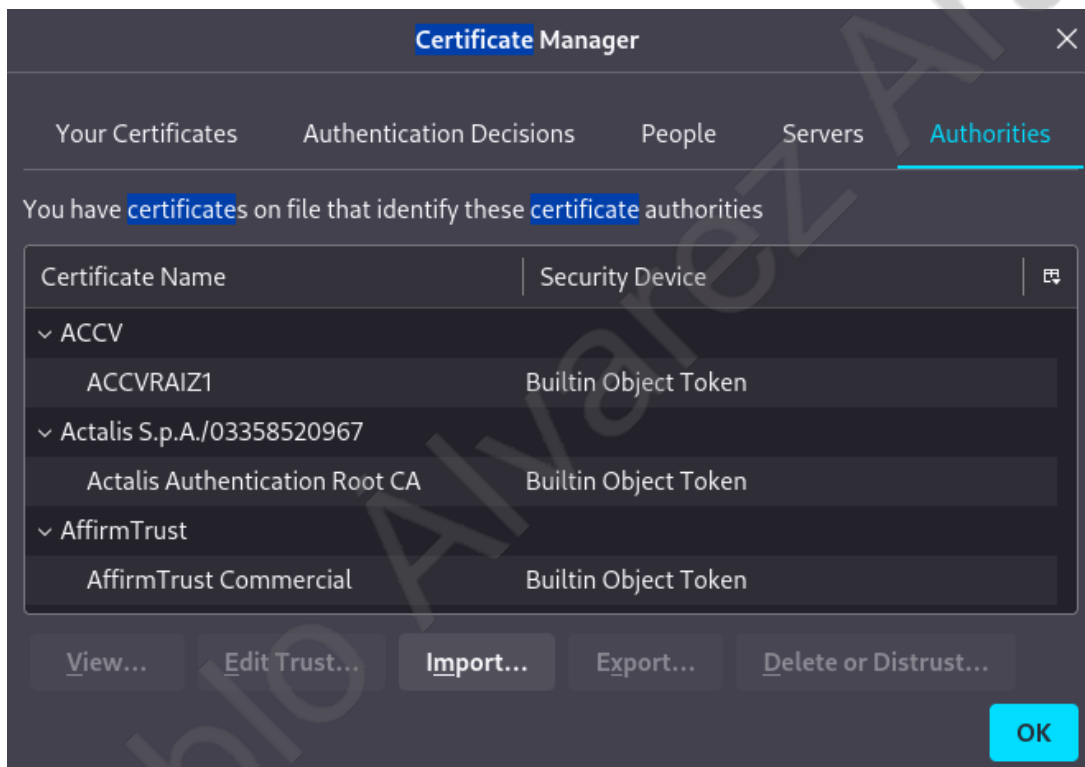


Vemos que se ha exportado correctamente y le damos a Close.

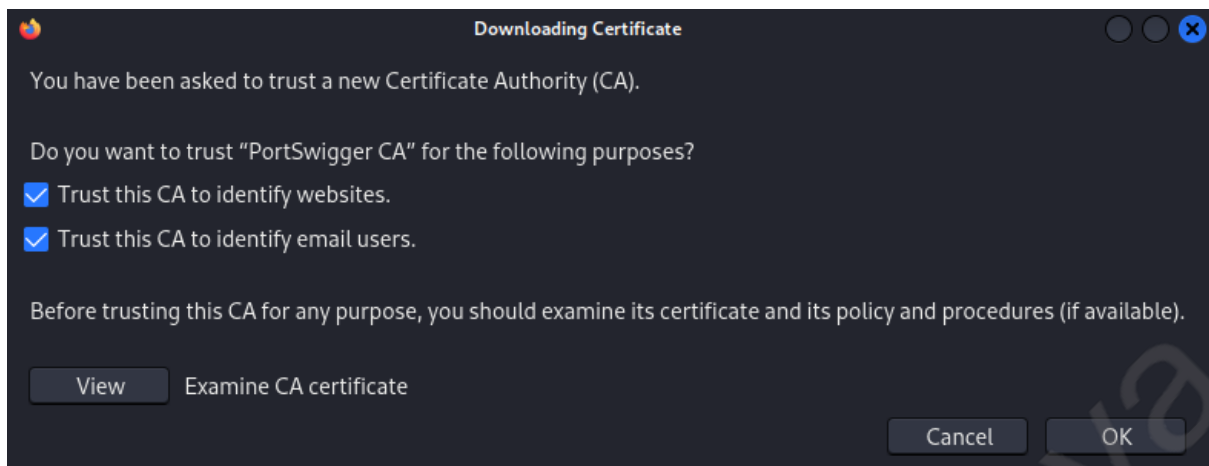
Paso 7:



Volvemos a la configuración del navegador, buscamos certificates y pulsamos en View Certificates...

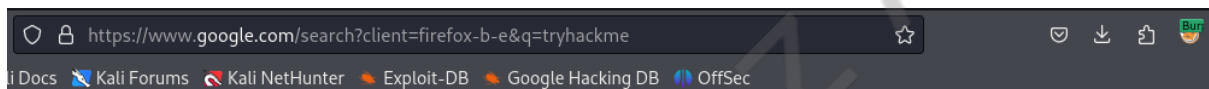


Se nos abre la pestaña de Certificate Manager donde le daremos en Import...



Al seleccionar nuevamente el certificado se nos abrirá una nueva pestaña en la que habilitaremos ambas opciones y pulsaremos OK.

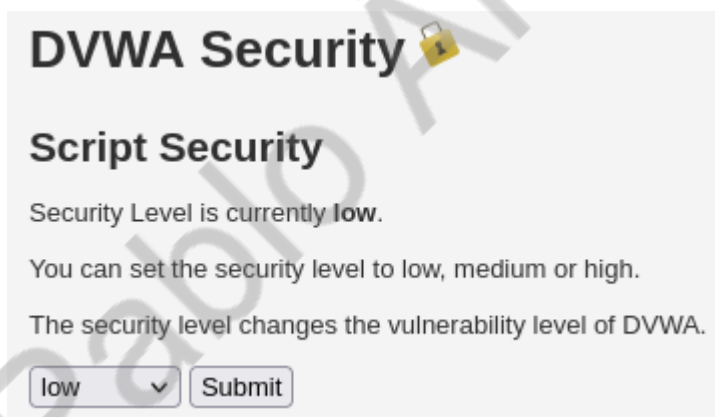
Luego cerramos también la pestaña Certificate Manager pulsando OK.



Y ya podemos acceder a sitios con protocolo HTTPS con Burp Suite activado.

Uso de Intruder: Sniper

Paso 1:



Iremos a la DVWA e iniciaremos con un nivel de seguridad bajo.

Paso 2:

Probaremos un ataque de fuerza bruta sobre el siguiente Log In teniendo el usuario correcto pero no la contraseña

Vulnerability: Brute Force

Login

Username:

Password:

Paso 3:

The screenshot shows the Burp Suite interface with the 'Proxy' tab selected. The 'Intercept' sub-tab is active, showing a request to `http://10.0.2.7:80`. The request is in 'Pretty' view, displaying the following details:

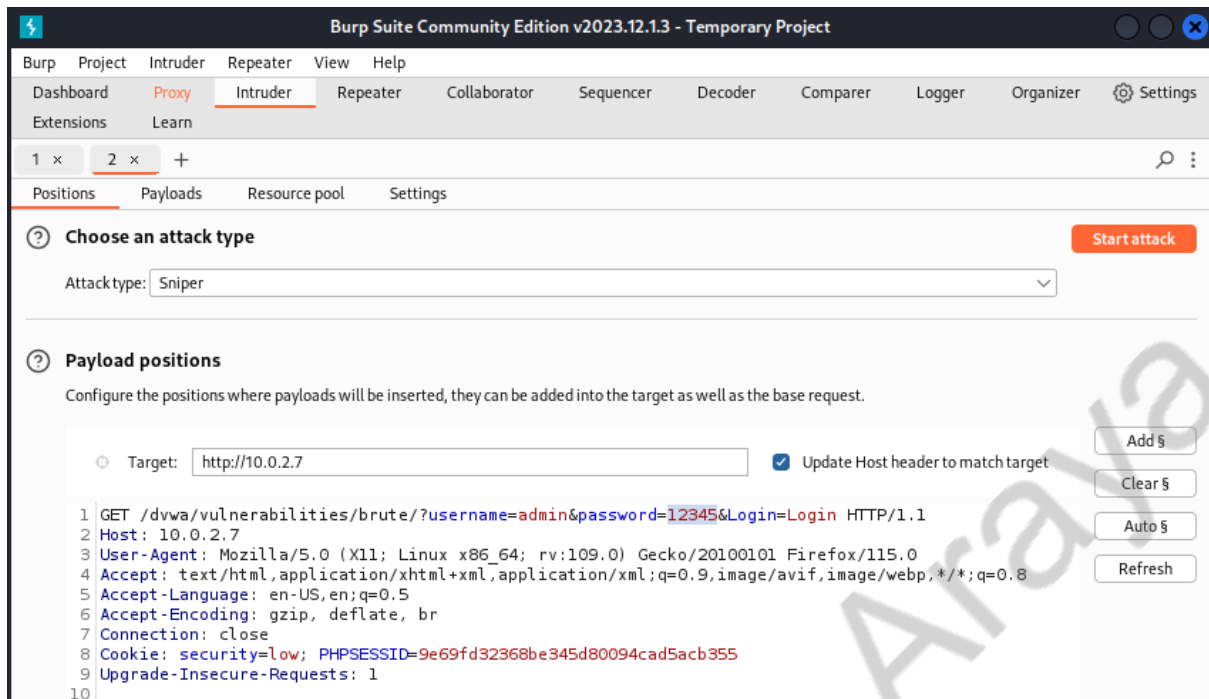
- 1 GET /dvwa/vulnerabilities/brute/?username=admin&password=12345&Login=Login HTTP/1.1
- 2 Host: 10.0.2.7
- 3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
- 4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
- 5 Accept-Language: en-US,en;q=0.5
- 6 Accept-Encoding: gzip, deflate, br
- 7 Connection: close
- 8 Referer: http://10.0.2.7/dvwa/vulnerabilities/brute/
- 9 Cookie: security=low; PHPSESSID=9e69fd32368be345d80094cad5acb355
- 10 Upgrade-Insecure-Requests: 1
- 11

On the right side, there is a context menu with the following options:

- Scan
- Scan selected insertion point
- Send to Intruder (Ctrl+I)

Nos movemos a Burp Suite y a la solicitud de la imagen anterior le damos clic derecho y luego pulsamos en Send to Intruder o Ctrl+I

Paso 4:



Vamos al Intruder, pintamos el valor de password y pulsamos en Add.

```
GET /dvwa/vulnerabilities/brute/?username=admin&password=$12345$&Login=Login HTTP/1.1
```

Y se nos añaden los caracteres que se muestran en la imagen.

Paso 5:

Positions

Payloads

Resource pool

Settings

?

Payload sets

You can define one or more payload sets. The number of payload sets depends on the attack each payload set, and each payload type can be customized in different ways.

Payload set: 1

Payload count: 5

Payload type: Simple list

Request count: 5

?

Payload settings [Simple list]

This payload type lets you configure a simple list of strings that are used as payloads.

Paste

Load ...

Remove

Clear

Deduplicate

pablo

andres

alvarez

araya

password

Add

Add from list ... [Pro version only]

Nos movemos a payload y agregamos manualmente los valores de nuestra Simple list

Start attack

Y presionamos en Start attack

Requ...	Payload	Status code	Error	Timeout	Length	Comment
0		200	<input type="checkbox"/>	<input type="checkbox"/>	4920	
1	pablo	200	<input type="checkbox"/>	<input type="checkbox"/>	4919	
2	andres	200	<input type="checkbox"/>	<input type="checkbox"/>	4920	
3	alvarez	200	<input type="checkbox"/>	<input type="checkbox"/>	4919	
4	araya	200	<input type="checkbox"/>	<input type="checkbox"/>	4920	
5	password	200	<input type="checkbox"/>	<input type="checkbox"/>	4985	

Request	Response
Pretty	Raw Hex Render
57	
58	<form action="#" method="GET">
59	Username:
	<input type="text" name="username">
60	Password:
	<input type="password" AUTOCOMPLETE="off" name="password">
61	<input type="submit" value="Login" name="Login">
62	</form>
63	
64	<p>
	Welcome to the password protected area admin
	</p>
65	
66	</div>
--	

Si damos clic en el Length que sobresale porque no es ni 4920 ni 4919 podemos ver que su Response devuelve un mensaje "Welcome to the password protected area admin"

Requ...	Payload	Status code	Error	Timeout	Length	Comment
0		200	<input type="checkbox"/>	<input type="checkbox"/>	4920	
1	pablo	200	<input type="checkbox"/>	<input type="checkbox"/>	4919	
2	andres	200	<input type="checkbox"/>	<input type="checkbox"/>	4920	
3	alvarez	200	<input type="checkbox"/>	<input type="checkbox"/>	4919	
4	araya	200	<input type="checkbox"/>	<input type="checkbox"/>	4920	
5	password	200	<input type="checkbox"/>	<input type="checkbox"/>	4985	

Request		Response			
		Pretty	Raw	Hex	Render
		<pre> <input type="password" AUTOCOMPLETE="off" name="password">
 <input type="submit" value="Login" name="Login"> </form> </pre>			
61					
62					
63					
64		<pre> <pre>
 Username and/or password incorrect. </pre> </pre>			

A diferencia que si pulsamos cualquier otro que nos muestra un mensaje de "Username and/or password incorrect."

Paso 6:

En el caso de tener un Simple list mucho más extenso tendríamos que filtrar por mensaje incorrecto para que nos deje el payload correcto al final de la lista y poderlo identificar de la siguiente manera:

4	araya	200	<input type="checkbox"/>	<input type="checkbox"/>	4920
5	password	200	<input type="checkbox"/>	<input type="checkbox"/>	4985

Request		Response			
		Pretty	Raw	Hex	Render
		<pre> <input type="password" AUTOCOMPLETE="off" name="password">
 <input type="submit" value="Login" name="Login"> </form> </pre>			
61					
62					
63					
64		<pre> <pre>
 Username and/or password incorrect. </pre> </pre>			

Result #4	
Scan	
Send to Intruder	Ctrl+I
Send to Repeater	Ctrl+R
Send to Sequencer	
Send to Organizer	Ctrl+O
Send to Comparer (request)	
Send to Comparer (response)	
Show response in browser	
Request in browser	>
Generate CSRF PoC	
Add to site map	
Request item again	
Define extract grep from response	

Damos clic derecho en cualquiera de los payloads incorrectos de la lista y pulsamos en Define extract grep from response.

Define extract grep item

Define the location of the item to be extracted. Selecting the item in the response panel will create a suitable configuration automatically. You can also modify the configuration manually to ensure it works effectively.

Define start and end

Start after expression:

<pre>Username and/or password incorrect.

Start at offset:

3555

End at delimiter:

</pre>

End at fixed length:

35

Extract from regex group

<pre>
{,?}</pre>

Case sensitive

Exclude HTTP headers

Update config based on selection below

Refetch response

56<h2>Login</h2>

57

58<form action="#" method="GET">

59Username:
<input type="text" name="username">

60Password:
<input type="password" AUTOCOMPLETE="off" name="password">

61<input type="submit" value="Login" name="Login">

62</form>

63

64<pre>
Username and/or password incorrect.</pre>

65

66</div>

67

68<h2>More info</h2>

69

70http://hiderefer.com/?http://www.owasp.org/index.php/Testing_for_Brute_Force_%28OWASP-AT-004%29" target="_blank" rel="noopener">

Username and/or password incorrect.

1 match

OKCancel

Ingresamos el mensaje en el buscador y luego también en el input de Start after expression entre sus correspondientes etiquetas html <pre>
 y presionamos en OK.

Paso 7:

Request	Payload	Status code	Error	Timeout	Length	Comment	<pre>Username and/or password incor...
0		200	<input type="checkbox"/>	<input type="checkbox"/>	4920		
1	pablo	200	<input type="checkbox"/>	<input type="checkbox"/>	4919		
2	andres	200	<input type="checkbox"/>	<input type="checkbox"/>	4920		
3	alvarez	200	<input type="checkbox"/>	<input type="checkbox"/>	4919		
4	araya	200	<input type="checkbox"/>	<input type="checkbox"/>	4920		
5	password	200	<input type="checkbox"/>	<input type="checkbox"/>	4985		

Así filtramos los resultados de tal forma que el payload correcto siempre se muestre al final de la lista.

Uso de Intruder: Battery Ram

Paso 1:

```
(root@kali)-[~]
# cd /usr/share/wordlists

(root@kali)-[/usr/share/wordlists]
# ls
amass             fern-wifi         rockyou.txt.gz
dirb              john.lst         sqlmap.txt
dirbuster         legion           wfuzz
dnsmap.txt        metasploit       wifite.txt
fasttrack.txt     nmap.lst
```

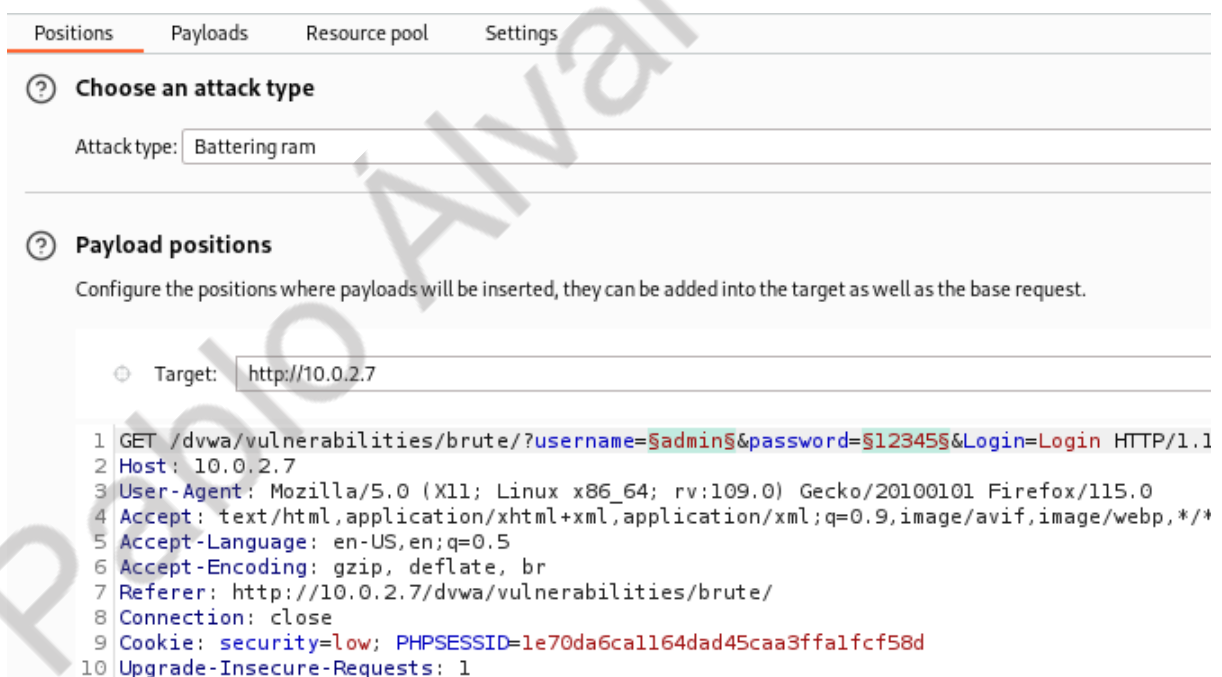
En kali linux existe un directorio con algunos wordlists que trae por defecto.

Comando: `cd /usr/share/wordlists`

Comando: `ls`

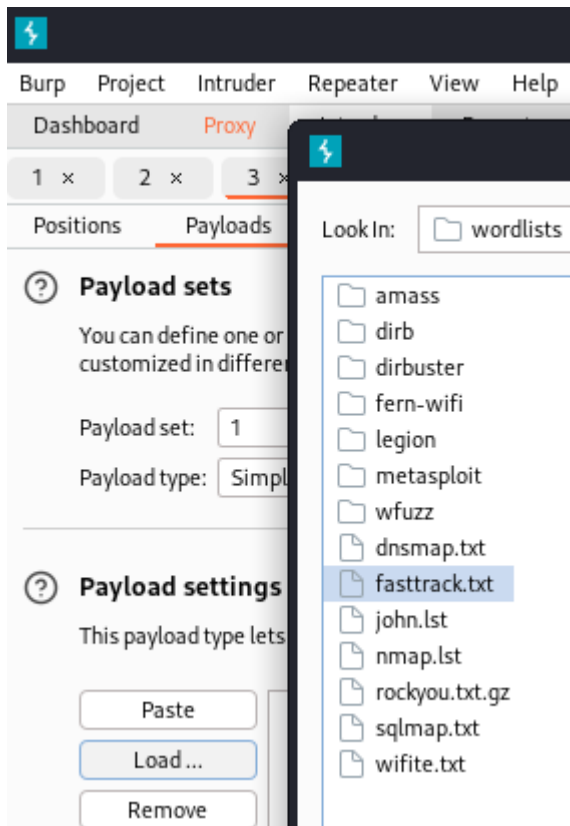
Paso 2:

Volvemos al Log In con la intención de que Burp Suite intercepte nuestra petición la cual enviaremos al Intruder desde donde seleccionaremos el tipo de ataque que esta vez será Battery Ram



Donde pintaremos tanto el valor del username como el de password y pulsaremos en Add.

Paso 3:



Nos movemos de Positions a Payloads y cargaremos fasttrack.txt

Paso 4:

Requ...	Payload	Status code	Error	Timeout	Length	Comment
6	spring2017	200	<input type="checkbox"/>	<input type="checkbox"/>	4919	
7	spring2016	200	<input type="checkbox"/>	<input type="checkbox"/>	4920	
8	spring2015	200	<input type="checkbox"/>	<input type="checkbox"/>	4919	
9	spring2014	200	<input type="checkbox"/>	<input type="checkbox"/>	4920	
10	spring2013	200	<input type="checkbox"/>	<input type="checkbox"/>	4919	
11	Summer2017	200	<input type="checkbox"/>	<input type="checkbox"/>	4919	
12	Summer2016	200	<input type="checkbox"/>	<input type="checkbox"/>	4919	
13	Summer2015	200	<input type="checkbox"/>	<input type="checkbox"/>	4920	
14	Summer2014	200	<input type="checkbox"/>	<input type="checkbox"/>	4919	
15	Summer2013	200	<input type="checkbox"/>	<input type="checkbox"/>	4919	
16	summer2017	200	<input type="checkbox"/>	<input type="checkbox"/>	4920	
17	summer2016	200	<input type="checkbox"/>	<input type="checkbox"/>	4919	
18	summer2015	200	<input type="checkbox"/>	<input type="checkbox"/>	4920	

Request	Response
<pre> 1 GET /dvwa/vulnerabilities/brute/?username=Summer2017&password=Summer2017&Login=Login HTTP/1.1 2 Host: 10.0.2.7 3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0 4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8 5 Accept-Language: en-US,en;q=0.5 6 Accept-Encoding: gzip, deflate, br 7 Referer: http://10.0.2.7/dvwa/vulnerabilities/brute/ 8 Connection: keep-alive 9 Cookie: security=low; PHPSESSID=1e70da6ca1164dad45caa3ffa1fcf58d 10 Upgrade-Insecure-Requests: 1 </pre>	

Pausamos el ataque con el fin de demostrar que Battery Ram lo que hace es probar el mismo valor tanto para el usuario como para la contraseña, que si bien no aplica para este caso, pudiera servir en un caso donde tanto el usuario y la password fueran iguales.

Uso de Intruder: Pitchfork

Paso 1:

Choose an attack type

Attack type:

Payload positions

Configure the positions where payloads will be inserted, they can be added into the target as well as the base request.

Target:

```
1 GET /dvwa/vulnerabilities/brute/?username=$admin$&password=$12345$&Login=Login HTTP/1.1
2 Host: 10.0.2.7
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate, br
7 Connection: close
8 Cookie: security=low; PHPSESSID=1e70da6ca1164dad45caa3ffa1fcf58d
9 Upgrade-Insecure-Requests: 1
```

Repetimos el proceso anterior pero esta vez seleccionamos Pitchfork como tipo de ataque, pintamos los valores y pulsamos en Add.

Paso 2:

Positions	Payloads	Resource pool	Settings
Payload sets			
You can define one or more payload sets. The number of payload sets d customized in different ways.			
Payload set:	<input type="text" value="1"/>	Payload count:	0
Payload type:	<div><div>1</div><div>2</div></div>	Request count:	0

Vemos que tenemos dos opciones para payload, esto se debe al tipo de ataque que vamos a efectuar, que a diferencia del Battery Ram nos permite cargar un sample list para username y uno para password.

Paso 3:

? **Payload sets**

You can define one or more payload sets. The number of payload sets depends on the a customized in different ways.

Payload set: 1 Payload count: 225

Payload type: Simple list Request count: 0

? **Payload settings [Simple list]**

This payload type lets you configure a simple list of strings that are used as payloads.

Paste

Load ...

Remove

Clear

Deduplicate

Add

Add from list ... [Pro version only]

pablo

andres

admin

Spring2017

Spring2016

Spring2015

Spring2014

Spring2013

spring2017

spring2016

Cargamos una simple list para el payload 1.

? **Payload sets**

You can define one or more payload sets. The number of payload sets depends on the al customized in different ways.

Payload set: 2 Payload count: 225

Payload type: Simple list Request count: 225

? **Payload settings [Simple list]**

This payload type lets you configure a simple list of strings that are used as payloads.

Paste

Load ...

Remove

Clear

Deduplicate

Add

Add from list ... [Pro version only]

andres

araya

password

Spring2017

Spring2016

Spring2015

Spring2014

Spring2013

spring2017

spring2016

Y una simple list para el payload 2.

Paso 4:

Requ...	Payload1	Payload2	Status code	Error	Timeout	Length	Comment
0			200	<input type="checkbox"/>	<input type="checkbox"/>	4920	
1	pablo	andres	200	<input type="checkbox"/>	<input type="checkbox"/>	4919	
2	andres	araya	200	<input type="checkbox"/>	<input type="checkbox"/>	4920	
3	admin	password	200	<input type="checkbox"/>	<input type="checkbox"/>	4985	
4	Spring2017	Spring2017	200	<input type="checkbox"/>	<input type="checkbox"/>	4920	
5	Spring2016	Spring2016	200	<input type="checkbox"/>	<input type="checkbox"/>	4919	
6	Spring2015	Spring2015	200	<input type="checkbox"/>	<input type="checkbox"/>	4920	
7	Spring2014	Spring2014	200	<input type="checkbox"/>	<input type="checkbox"/>	4920	
8	Spring2013	Spring2013	200	<input type="checkbox"/>	<input type="checkbox"/>	4920	
9	spring2017	spring2017	200	<input type="checkbox"/>	<input type="checkbox"/>	4919	
10	spring2016	spring2016	200	<input type="checkbox"/>	<input type="checkbox"/>	4920	
11	spring2015	spring2015	200	<input type="checkbox"/>	<input type="checkbox"/>	4919	
12	spring2014	spring2014	200	<input type="checkbox"/>	<input type="checkbox"/>	4920	

Request

Response

Pretty

Raw

Hex

1 GET /dvwa/vulnerabilities/brute/?username=admin&password=password&Login=Login HTTP/1.1

2 Host: 10.0.2.7

3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0

4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8

5 Accept-Language: en-US,en;q=0.5

6 Accept-Encoding: gzip, deflate, br

7 Connection: keep-alive

8 Cookie: security=low; PHPSESSID=1e70da6ca1164dad45caa3ffa1fcf58d

9 Upgrade-Insecure-Requests: 1

Podemos ver que nos ha dado un resultado positivo, sin embargo, esto ha sido a modo demostrativo puesto que tanto admin como password se encuentran en la tercera posición en sus correspondientes wordlists y de no ser el caso, esto no habría sucedido.

Uso de Intruder: Cluster bomb

Paso 1:

Choose an attack type

Attack type: Cluster bomb

Payload positions

Configure the positions where payloads will be inserted, they can be added into the target as well as the base request.

Target: http://10.0.2.7

1 GET /dvwa/vulnerabilities/brute/?username=\$admin\$&password=\$12345\$&Login=Login HTTP/1.1

2 Host: 10.0.2.7

3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0

4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*

5 Accept-Language: en-US,en;q=0.5

6 Accept-Encoding: gzip, deflate, br

7 Connection: close

8 Cookie: security=low; PHPSESSID=1e70da6ca1164dad45caa3ffa1fcf58d

9 Upgrade-Insecure-Requests: 1

Repetimos los mismos pasos anteriores seleccionando esta vez el tipo de ataque Cluster bomb.

Paso 2:

? **Payload sets**

You can define one or more payload sets. The number of payload sets depends on the customized in different ways.

Payload set: Payload count: 3

Payload type: Request count: 0

? **Payload settings [Simple list]**

This payload type lets you configure a simple list of strings that are used as payloads.

admin
user
administrator

Cargamos nuestra Simple list para el payload 1.

? **Payload sets**

You can define one or more payload sets. The number of payload sets depends on the customized in different ways.

Payload set: Payload count: 3

Payload type: Request count: 9

? **Payload settings [Simple list]**

This payload type lets you configure a simple list of strings that are used as payloads.

pass
paswd
password

Y nuestra Simple list para el payload 2.

Paso 3:

Requ...	Payload 1	Payload 2	Status code	Error	Timeout	Length	Comment
0			200	<input type="checkbox"/>	<input type="checkbox"/>	4920	
1	admin	pass	200	<input type="checkbox"/>	<input type="checkbox"/>	4919	
2	user	pass	200	<input type="checkbox"/>	<input type="checkbox"/>	4920	
3	administrator	pass	200	<input type="checkbox"/>	<input type="checkbox"/>	4919	
4	admin	passwd	200	<input type="checkbox"/>	<input type="checkbox"/>	4920	
5	user	passwd	200	<input type="checkbox"/>	<input type="checkbox"/>	4919	
6	administrator	passwd	200	<input type="checkbox"/>	<input type="checkbox"/>	4920	
7	admin	password	200	<input type="checkbox"/>	<input type="checkbox"/>	4985	
8	user	password	200	<input type="checkbox"/>	<input type="checkbox"/>	4920	
9	administrator	password	200	<input type="checkbox"/>	<input type="checkbox"/>	4919	

Request	Response
<div> <div>Pretty</div> <div>Raw</div> <div>Hex</div> </div> <pre> 1 GET /dwwa/vulnerabilities/brute/?username=admin&password=password&Login=Login HTTP/1.1 2 Host: 10.0.2.7 3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0 4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8 5 Accept-Language: en-US,en;q=0.5 6 Accept-Encoding: gzip, deflate, br 7 Connection: keep-alive 8 Cookie: security=low; PHPSESSID=1e70da6ca1164dad45caa3ffa1fcf58d 9 Upgrade-Insecure-Requests: 1 </pre>	

Podemos observar que el ataque ha salido exitoso sin importar la posición de los valores a probar en nuestros diferentes payloads, esto se debe a que Cluster bomb compara todas las combinaciones posibles, sin embargo, este tipo de ataque suele demorarse más.

Uso de Repeater

Paso 1:

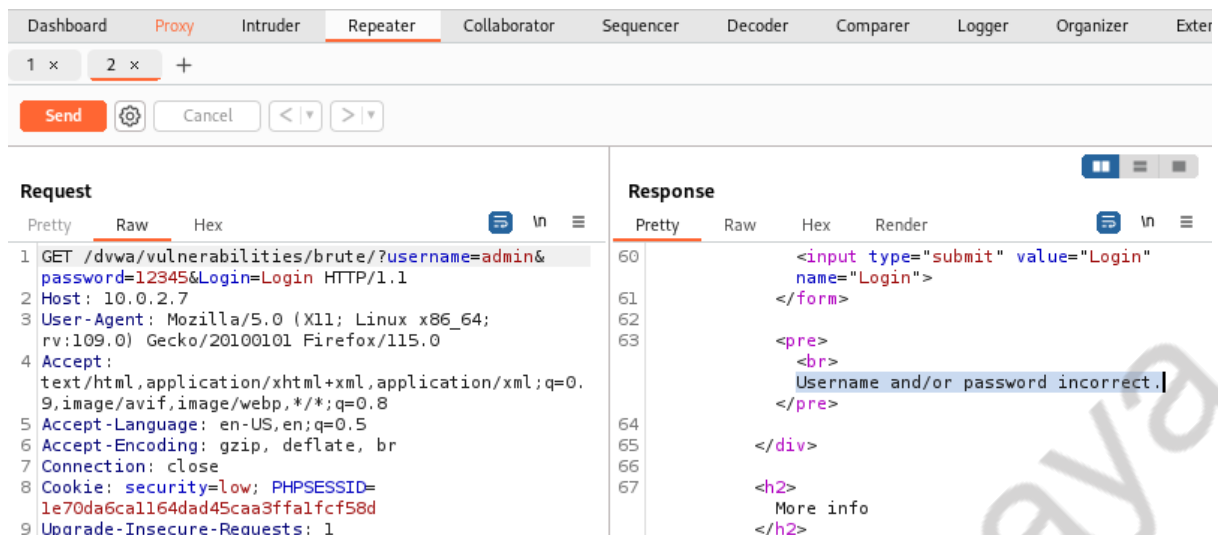
Intercept HTTP history WebSockets history Proxy settings

Filter settings: Hiding CSS, image and general binary content

# ^	Host	Method	URL	Params
1	http://10.0.2.7	GET	/dwwa/vulnerabilities/brute/?username=...	✓
2	https://firefox.settings.services....	GET	/v1/buckets/main/collectio	http://10.0.2.7/dwwa/
3	http://10.0.2.7	GET	/dwwa/vulnerabilities/brut	Add to scope
4	http://10.0.2.7	GET	/dwwa/vulnerabilities/brut	Scan
6	https://contile.services.mozilla.c...	GET	/v1/tiles	Send to Intruder
7	http://10.0.2.7	GET	/dwwa/vulnerabilities/brut	Send to Repeater
8	https://safebrowsing.googleapis...	GET	/v4/threatListUpdates:fet	

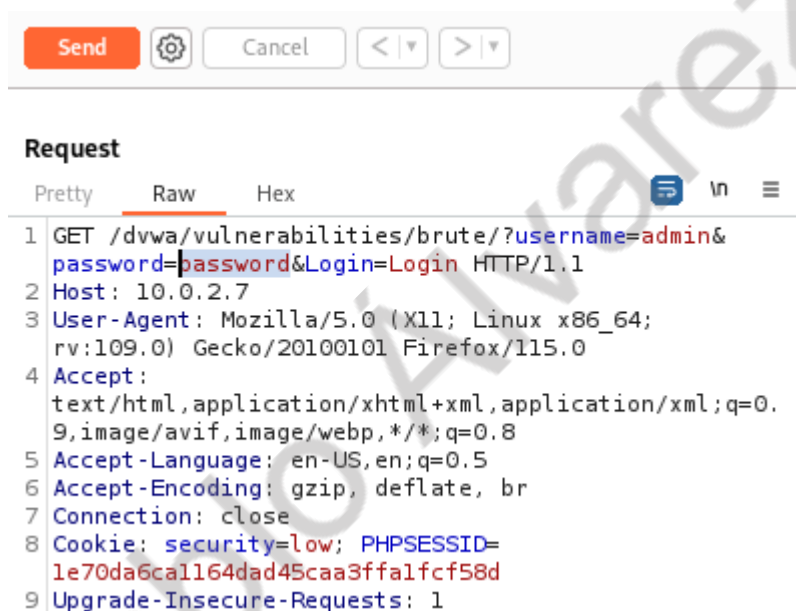
Nos ubicamos en HTTP history, damos clic derecho en alguno de los registros y pulsamos Send to Repeater.

Paso 2:



Nos vamos a Repeater y pulsamos en Send podremos observar tanto la Consulta como la Respuesta de la solicitud.

Paso 3:



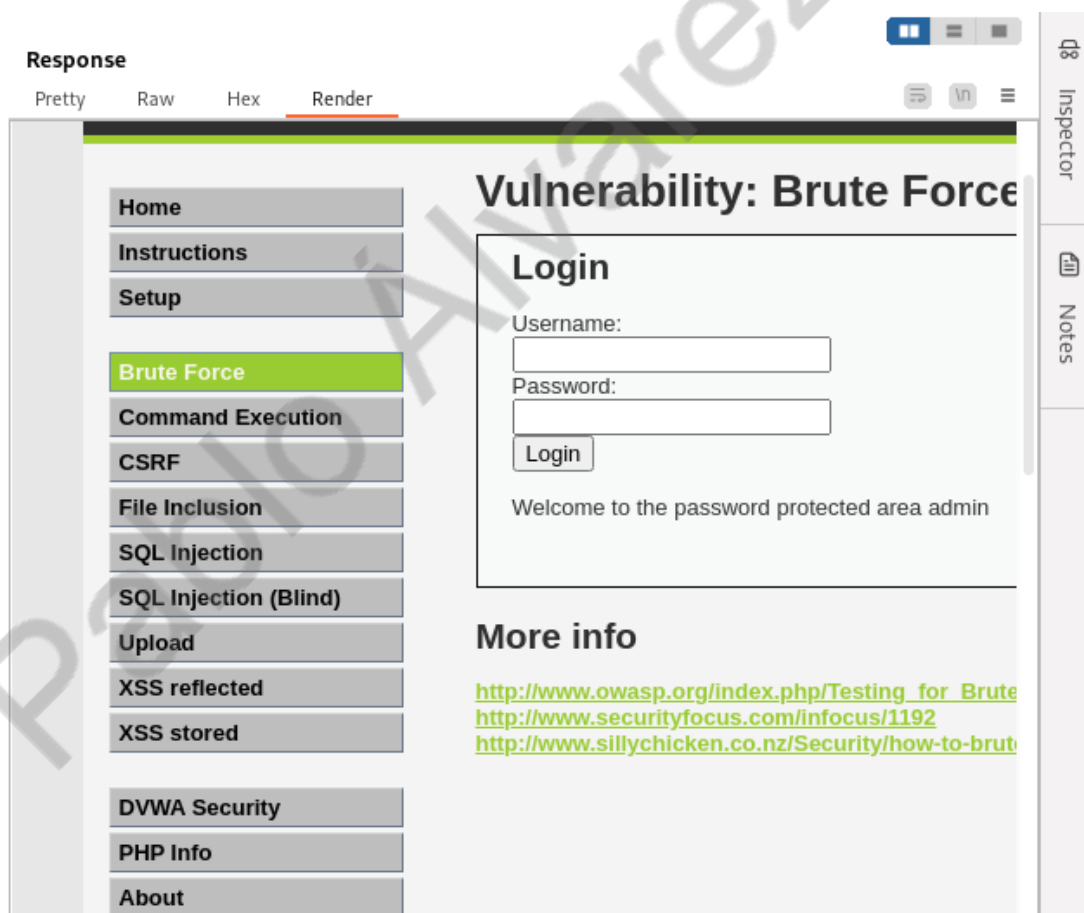
Si modificamos los valores de los parámetros de nuestro Request y presionamos en Send, Burp Suite nos mostrará de manera inmediata la respuesta del lado del Request como se puede apreciar en la siguiente imagen:

Response

Pretty Raw Hex Render

```
60 <input type="password" AUTOCOMPLETE="
61 off" name="password">
62 <br>
63 <input type="submit" value="Login"
   name="Login">
   </form>
   <p>
     Welcome to the password protected area
     admin
   </p>
   
66
67 </div>
   <h2>
     More info
   </h2>
```

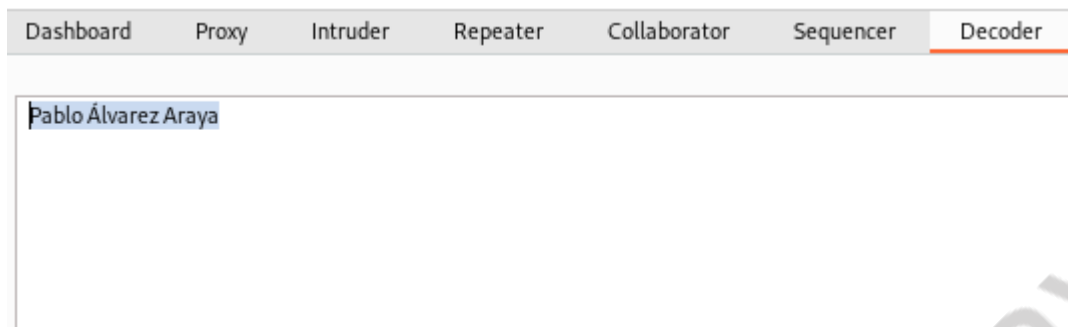
Paso 4:



Si dentro del mismo Response damos clic en Render, podremos ver la respuesta de una forma más rápida como si se tratase del navegador.

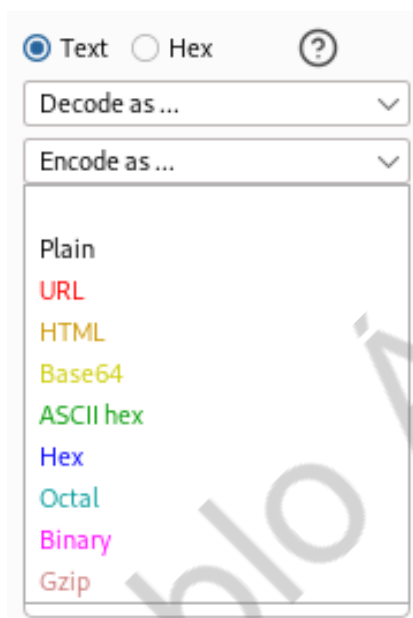
Uso del Decoder

Paso 1:



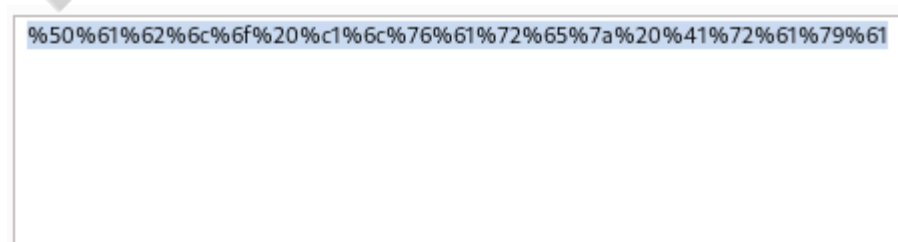
Nos dirigimos a Decoder y escribimos cualquier cadena de texto plano, en este caso mi nombre

Paso 2:



Si desplegamos las opciones que se muestran en Encode as ...

Y pulsamos cualquiera de ellas, en este caso yo he escogido la que dice URL, Burp Suite cambiará lo que hayamos escrito a dicha codificación como se muestra en la siguiente imagen:



Uso del Comparer

Paso 1:

The screenshot shows the Burp Suite Repeater tab. At the top, there are tabs for Dashboard, Proxy, Intruder, Repeater (selected), Collaborator, Sequencer, Decoder, and Comparer. Below these are buttons for 3 x, 4 x, 5 x, and a plus sign. A 'Send' button is highlighted in orange. Below the buttons are navigation arrows and a 'Cancel' button. The main area is split into 'Request' and 'Response' sections. The 'Request' section has tabs for Pretty, Raw, and Hex. The 'Response' section has tabs for Pretty, Raw, Hex, and Render. A context menu is open over the 'Response' section, showing options: Scan, Send to Intruder (Ctrl+I), Send to Repeater (Ctrl+R), Send to Sequencer, Send to Comparer, and Send to Decoder. The 'Request' text is: 1 GET /dvwa/vulnerabilities/brute/?username=admin&password=password&Login=Login HTTP/1.1 2 Host: 10.0.2.7 3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0 4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8 5 Accept-Language: en-US,en;q=0.5 6 Accept-Encoding: gzip, deflate, br. The 'Response' text is: 1 HTTP/1.1 200 OK 2 Date: Sun, 10 Mar 2024 09:51:01 GMT 3 Server: Apache/2.2.8 (Ubuntu) DAV/2 4 X-Powered-By: PHP/5.3.4-1ubuntu5.10 5 Pragma: no-cache 6 Cache-Control: no-cache, no-store, max-age=0, must-revalidate 7 Expires: 0 8 Connection: close 9 Content-Type: text/html; charset=UTF-8 10 11 12 13 <!DOCTYPE html>

Seguimos en el Repeater y damos clic derecho en Response y pulsamos Send to Comparer.

Paso 2:

The screenshot shows the Burp Suite Comparer tab. At the top, there are tabs for Dashboard, Proxy, Intruder, Repeater, Collaborator, Sequencer, Decoder, Comparer (selected), and Settings. Below these are buttons for Logger, Organizer, Extensions, and Learn. The main area is titled 'Comparer' and contains a description: 'This function lets you do a word- or byte-level comparison between different data. You can load, paste, or send data here from other tools and then select the comparison you want to perform.' Below the description is a table with the following data:

#	Length	Data
1	4948	HTTP/1.1 200 OKDate: Sun, 10 Mar 2024 09:51:01 GMT

To the right of the table are buttons for Paste, Load, Remove, and Clear.

Vamos al Comparer y vemos que se ha agregado nuestra solicitud en el ítem 1.

Paso 3:

Dashboard Proxy Intruder Repeater Collaborator Sequencer Decoder Comparer Logger

3 x 4 x 5 x +

Send Cancel < >

Request

```

1 GET /dvwa/vulnerabilities/brute/?username=admin&password=1234&Login=Login HTTP/1.1
2 Host: 10.0.2.7
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate, br

```

Response

```

60 <input type="submit" value="Login" name="Login">
61 </form>
62
63 <pre>
  Username and/or password incorrect.
</pre>

```

Volvemos al Repeater, modificamos el valor de password y presionamos Send para cambiar el Response.

Paso 4:

Response

Pretty Raw Hex Render

```

60 <input type="submit" value="Login" name="Login">
61
62
63

```

Scan

- Send to Intruder Ctrl+I
- Send to Repeater Ctrl+R
- Send to Sequencer
- Send to Comparer

Este nuevo Response lo enviamos a Comparer.

Select item 1:

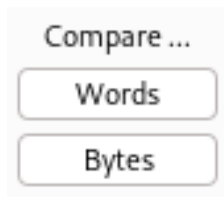
#	Length	Data
1	4948	HTTP/1.1 2...
2	4882	HTTP/1.1 2...

Select item 2:

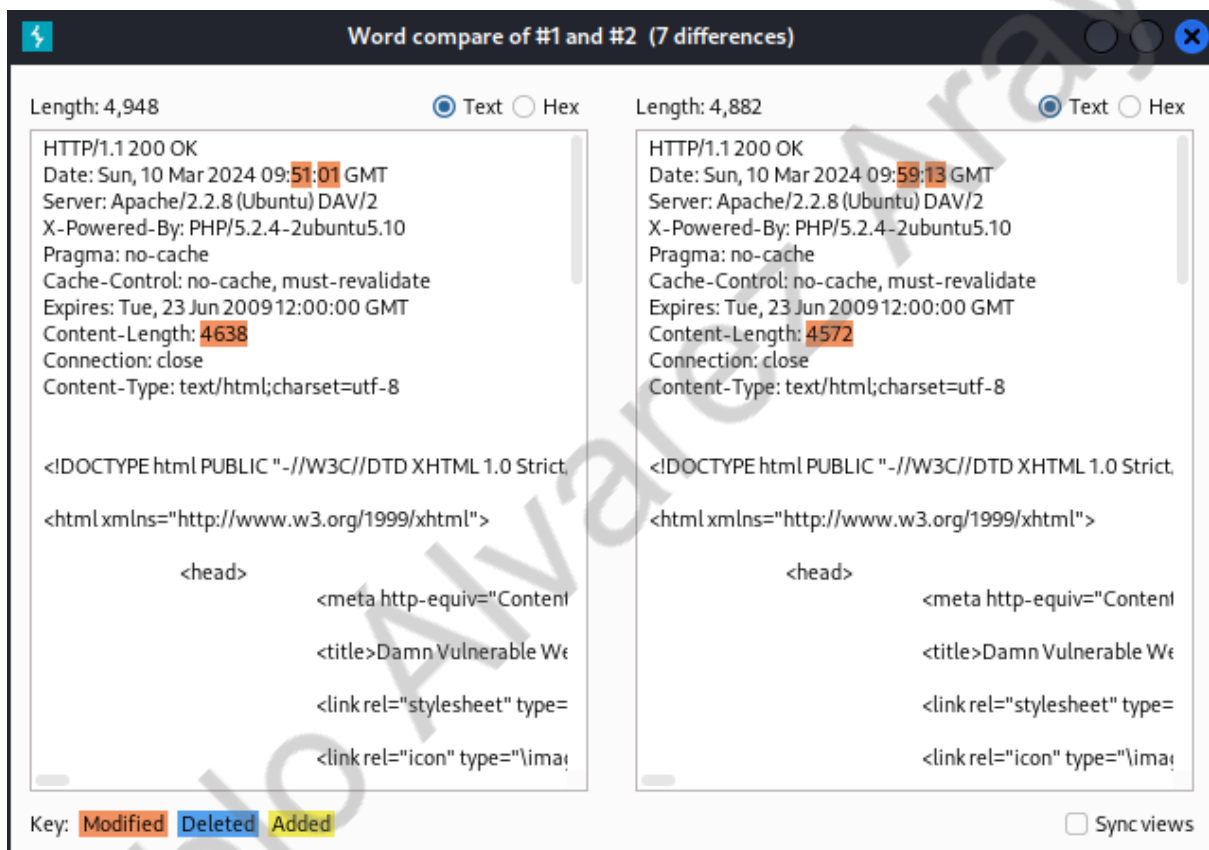
#	Length	Data
1	4948	HTTP/1.1 2...
2	4882	HTTP/1.1 2...

Vemos que se ha agregado nuestro Response en el segundo ítem.

Paso 5:



Presionamos el botón de Worlds ubicado en la esquina inferior derecha del programa.



Se nos abre una ventana que nos permite comparar ambas respuestas con una nomenclatura de colores y si pulsamos el check Sync views podremos hacer scroll de ambas respuestas al mismo tiempo.

Pablo Álvarez Araya