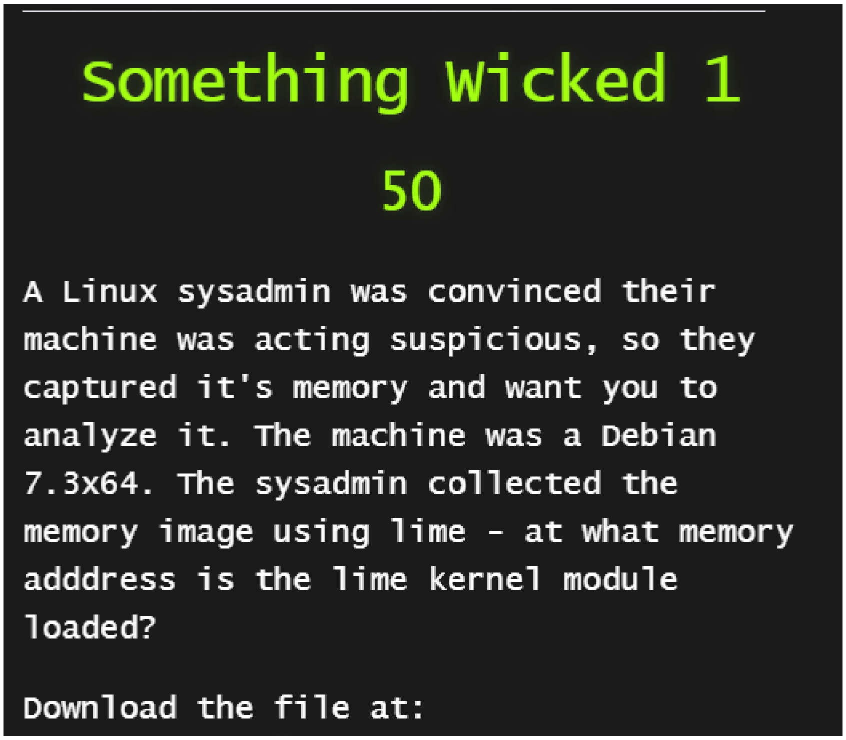


Something Wicked 1

50

A Linux sysadmin was convinced their machine was acting suspicious, so they captured it's memory and want you to analyze it. The machine was a Debian 7.3x64. The sysadmin collected the memory image using lime - at what memory adddress is the lime kernel module loaded?

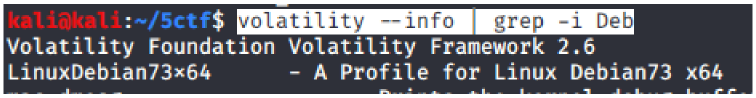
Download the file at:

First, we need to get a profile for this: https://github.com/volatilityfoundation/profiles/blob/master/Linux/Debian/x64/Debian73.zip

Once you get that profile downloaded, we need to move it over the the overlays/linux folder for volatility. On Kali it is: /usr/lib/python2.7/dist-packages/volatility/plugins/overlays/linux/

From the Downloads folder run this command to copy over: sudo cp Debian73.zip /usr/lib/python2.7/dist-packages/volatility/plugins/overlays/linux/

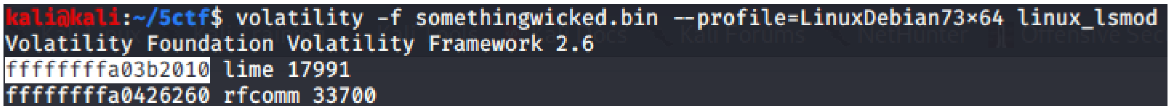
To verify the profile is loaded into Volility run: volatility --info | grep -i Deb



```
kali@kali:~/5ctf$ volatility --info | grep -i Deb
Volatility Foundation Volatility Framework 2.6
LinuxDebian73x64       - A Profile for Linux Debian73 x64
```

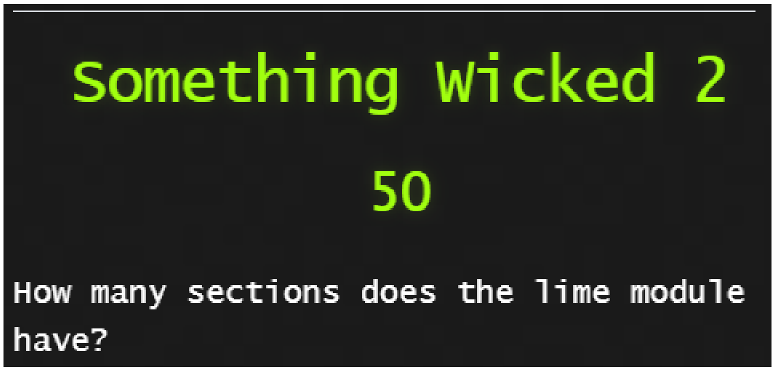
Run the linux_lsmod to get the location of kernel modules.

volatility -f somethingwicked.bin --profile=LinuxDebian73x64 linux_lsmod



```
kali@kali:~/5ctf$ volatility -f somethingwicked.bin --profile=LinuxDebian73x64 linux_lsmod
Volatility Foundation Volatility Framework 2.6
ffffffffa03b2010 lime 17991
ffffffffa0426260 rfcomm 33700
```

Flag: ffffffffa03b2010



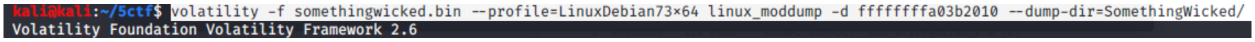
Something Wicked 2

50

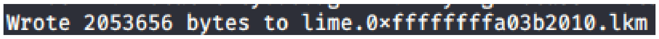
How many sections does the lime module have?

Now we want to dump the module out of the memory dump:

volatility -f somethingwicked.bin --profile=LinuxDebian73x64 linux_moddump -d ffffffffa03b2010 --dump-dir=SomethingWicked/



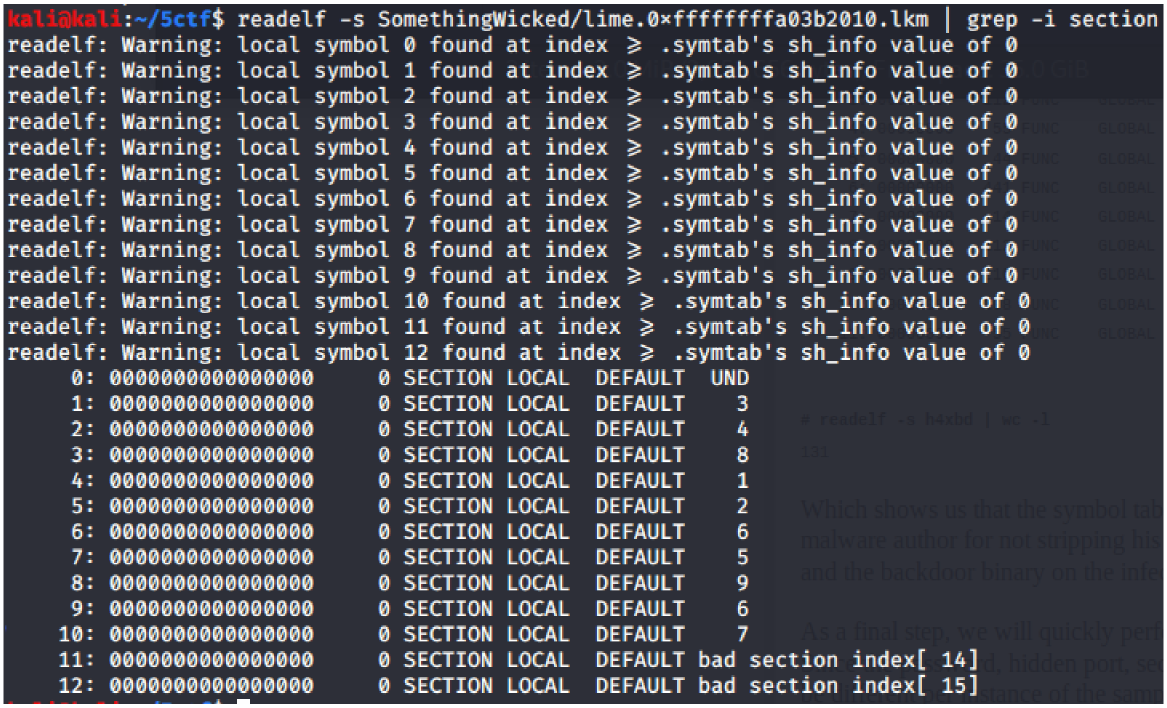
```
kali@kali:~/5ctf$ volatility -f somethingwicked.bin --profile=LinuxDebian73x64 linux_moddump -d ffffffffa03b2010 --dump-dir=SomethingWicked/
Volatility Foundation Volatility Framework 2.6
```



```
Wrote 2053656 bytes to lime.0xffffffffa03b2010.lkm
```

Then we can run a program to read the file as a elf program called readelf

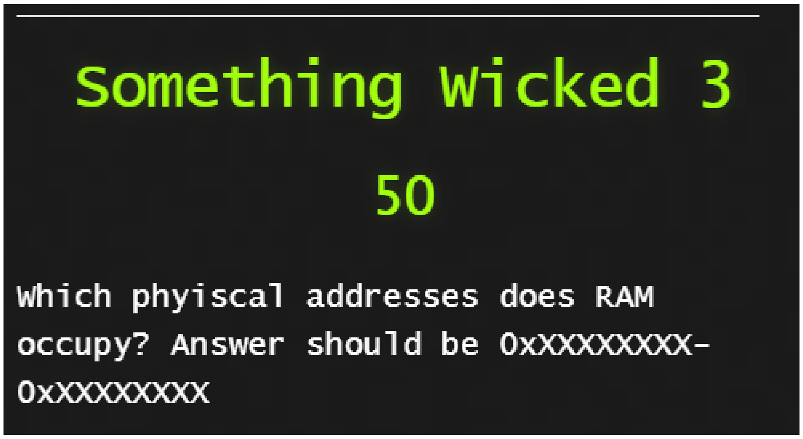
Readelf -s SomethingWicked\lime.0xffffffffa03b2010 | grep -I section



```
kali@kali:~/5ctf$ readelf -s SomethingWicked/lime.0xffffffffa03b2010.lkm | grep -i section
readelf: Warning: local symbol 0 found at index >= .symtab's sh_info value of 0
readelf: Warning: local symbol 1 found at index >= .symtab's sh_info value of 0
readelf: Warning: local symbol 2 found at index >= .symtab's sh_info value of 0
readelf: Warning: local symbol 3 found at index >= .symtab's sh_info value of 0
readelf: Warning: local symbol 4 found at index >= .symtab's sh_info value of 0
readelf: Warning: local symbol 5 found at index >= .symtab's sh_info value of 0
readelf: Warning: local symbol 6 found at index >= .symtab's sh_info value of 0
readelf: Warning: local symbol 7 found at index >= .symtab's sh_info value of 0
readelf: Warning: local symbol 8 found at index >= .symtab's sh_info value of 0
readelf: Warning: local symbol 9 found at index >= .symtab's sh_info value of 0
readelf: Warning: local symbol 10 found at index >= .symtab's sh_info value of 0
readelf: Warning: local symbol 11 found at index >= .symtab's sh_info value of 0
readelf: Warning: local symbol 12 found at index >= .symtab's sh_info value of 0
     0: 0000000000000000     0 SECTION LOCAL  DEFAULT  UND
     1: 0000000000000000     0 SECTION LOCAL  DEFAULT    3
     2: 0000000000000000     0 SECTION LOCAL  DEFAULT    4
     3: 0000000000000000     0 SECTION LOCAL  DEFAULT    8
     4: 0000000000000000     0 SECTION LOCAL  DEFAULT    1
     5: 0000000000000000     0 SECTION LOCAL  DEFAULT    2
     6: 0000000000000000     0 SECTION LOCAL  DEFAULT    6
     7: 0000000000000000     0 SECTION LOCAL  DEFAULT    5
     8: 0000000000000000     0 SECTION LOCAL  DEFAULT    9
     9: 0000000000000000     0 SECTION LOCAL  DEFAULT    6
    10: 0000000000000000     0 SECTION LOCAL  DEFAULT    7
    11: 0000000000000000     0 SECTION LOCAL  DEFAULT bad section index[ 14]
    12: 0000000000000000     0 SECTION LOCAL  DEFAULT bad section index[ 15]
```

We can see that there are 12 sections.

Flag: 12



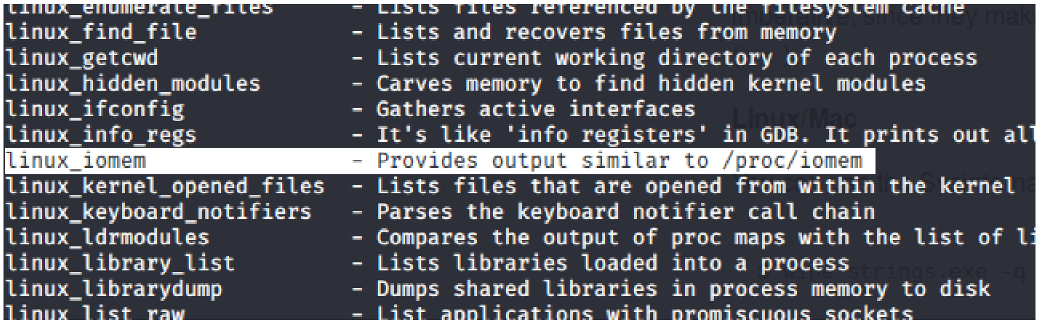
Something Wicked 3

50

Which phyiscal addresses does RAM occupy? Answer should be 0xXXXXXXXX-0xXXXXXXXX

Doing a list of all the Linux modules we can see what might be related to memory:

volatility --info | grep -i linux

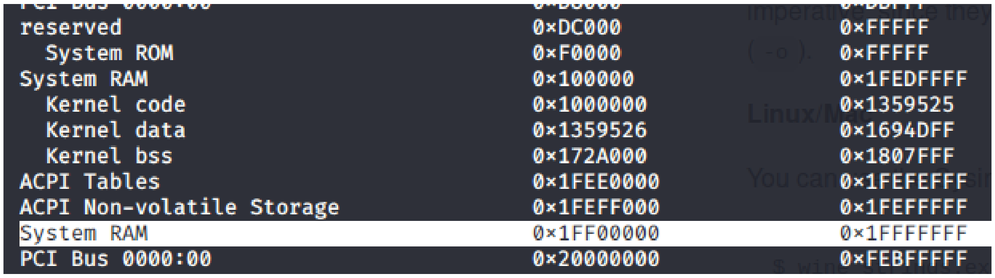


```
linux_enumerate_files        - Lists files referenced by the filesystem cache
linux_find_file              - Lists and recovers files from memory
linux_getcwd                 - Lists current working directory of each process
linux_hidden_modules         - Carves memory to find hidden kernel modules
linux_ifconfig               - Gathers active interfaces
linux_info_regs              - It's like 'info registers' in GDB. It prints out al
linux_iomem                  - Provides output similar to /proc/iomem
linux_kernel_opened_files    - Lists files that are opened from within the kernel
linux_keyboard_notifiers     - Parses the keyboard notifier call chain
linux_ldrmodules             - Compares the output of proc maps with the list of li
linux_library_list           - Lists libraries loaded into a process
linux_librarydump            - Dumps shared libraries in process memory to disk
linux_list_raw               - List applications with promiscuous sockets
```

Linux_iomem looks interesting:

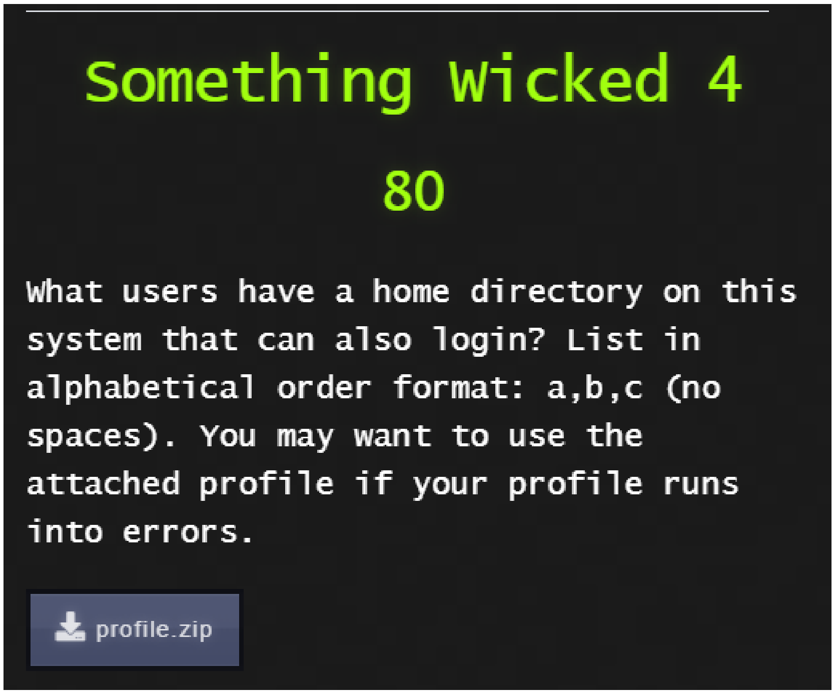
linux_iomem - Provides output similar to /proc/iomem

volatility -f somethingwicked.bin --profile=LinuxDebian73x64 linux_iomem



```
PCI Bus 0000:00                         0xD8000                 0xDBFFF
reserved                                0xDC000                 0xFFFFF
  System ROM                            0xF0000                 0xFFFFF
System RAM                              0x100000                0x1FEDFFFF
  Kernel code                           0x1000000               0x1359525
  Kernel data                           0x1359526               0x1694DFF
  Kernel bss                            0x172A000               0x1807FFF
ACPI Tables                             0x1FEE0000              0x1FEFEFFF
ACPI Non-volatile Storage               0x1FEFF000              0x1FEFFFFF
System RAM                              0x1FF00000              0x1FFFFFFF
PCI Bus 0000:00                         0x20000000              0xFEBFFFFF
```

Flag: 0x1FF00000-0x1FFFFFFF



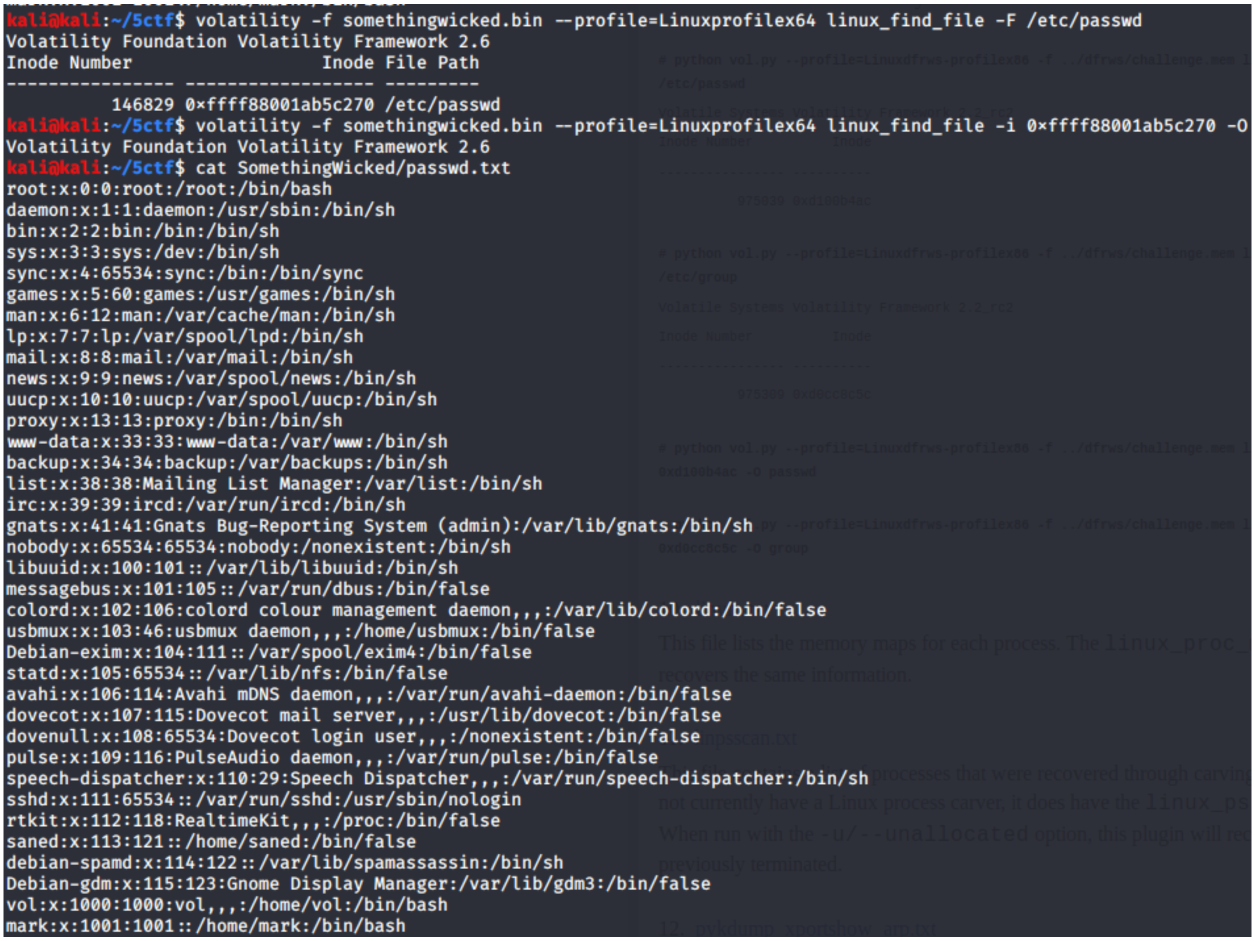
Something Wicked 4

80

What users have a home directory on this system that can also login? List in alphabetical order format: a,b,c (no spaces). You may want to use the attached profile if your profile runs into errors.

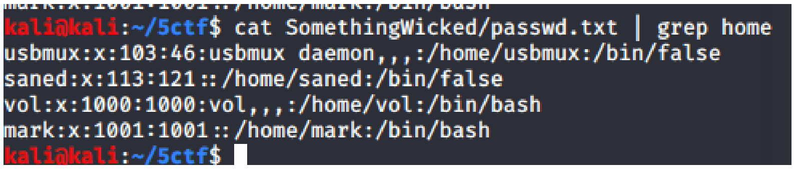
My profile did not find the file, so I had to use the profile.zip and copy it over to the same location as the other profile we used earlier

Now we will use the linux_find_file module to locate the file /etc/passwd and then grab the contents by the inode location that we get.



```
kali@kali:~/5ctf$ volatility -f somethingwicked.bin --profile=Linuxprofilex64 linux_find_file -F /etc/passwd
Volatility Foundation Volatility Framework 2.6
Inode Number                 Inode File Path
---------------- ------------------ ---------
          146829 0xffff88001ab5c270 /etc/passwd
kali@kali:~/5ctf$ volatility -f somethingwicked.bin --profile=Linuxprofilex64 linux_find_file -i 0xffff88001ab5c270 -O
Volatility Foundation Volatility Framework 2.6
kali@kali:~/5ctf$ cat SomethingWicked/passwd.txt
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/bin/sh
bin:x:2:2:bin:/bin:/bin/sh
sys:x:3:3:sys:/dev:/bin/sh
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/bin/sh
man:x:6:12:man:/var/cache/man:/bin/sh
lp:x:7:7:lp:/var/spool/lpd:/bin/sh
mail:x:8:8:mail:/var/mail:/bin/sh
news:x:9:9:news:/var/spool/news:/bin/sh
uucp:x:10:10:uucp:/var/spool/uucp:/bin/sh
proxy:x:13:13:proxy:/bin:/bin/sh
www-data:x:33:33:www-data:/var/www:/bin/sh
backup:x:34:34:backup:/var/backups:/bin/sh
list:x:38:38:Mailing List Manager:/var/list:/bin/sh
irc:x:39:39:ircd:/var/run/ircd:/bin/sh
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/bin/sh
nobody:x:65534:65534:nobody:/nonexistent:/bin/sh
libuuid:x:100:101::/var/lib/libuuid:/bin/sh
messagebus:x:101:105::/var/run/dbus:/bin/false
colord:x:102:106:colord colour management daemon,,,:/var/lib/colord:/bin/false
usbmux:x:103:46:usbmux daemon,,,:/home/usbmux:/bin/false
Debian-exim:x:104:111::/var/spool/exim4:/bin/false
statd:x:105:65534::/var/lib/nfs:/bin/false
avahi:x:106:114:Avahi mDNS daemon,,,:/var/run/avahi-daemon:/bin/false
dovecot:x:107:115:Dovecot mail server,,,:/usr/lib/dovecot:/bin/false
dovenull:x:108:65534:Dovecot login user,,,:/nonexistent:/bin/false
pulse:x:109:116:PulseAudio daemon,,,:/var/run/pulse:/bin/false
speech-dispatcher:x:110:29:Speech Dispatcher,,,:/var/run/speech-dispatcher:/bin/sh
sshd:x:111:65534::/var/run/sshd:/usr/sbin/nologin
rtkit:x:112:118:RealtimeKit,,,:/proc:/bin/false
saned:x:113:121::/home/saned:/bin/false
debian-spamd:x:114:122::/var/lib/spamassassin:/bin/sh
Debian-gdm:x:115:123:Gnome Display Manager:/var/lib/gdm3:/bin/false
vol:x:1000:1000:vol,,,:/home/vol:/bin/bash
mark:x:1001:1001::/home/mark:/bin/bash
```

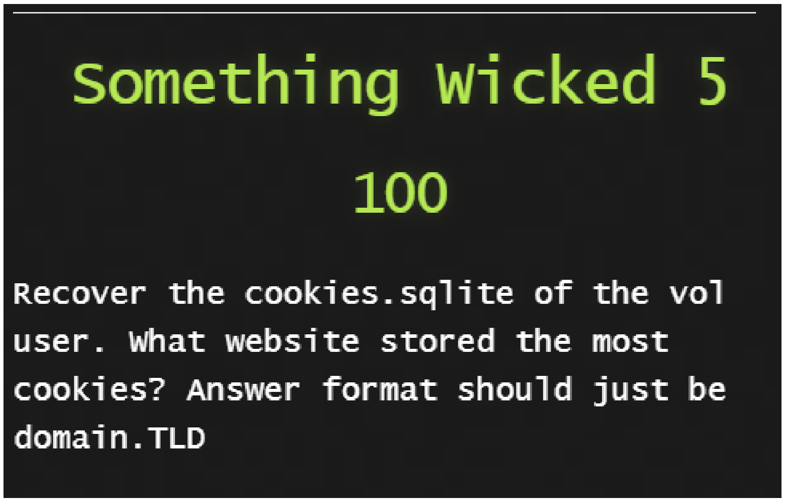
To limit this, we grep home and find 2 users with /bin/false and 2 users with /bin/bash



```
kali@kali:~/5ctf$ cat SomethingWicked/passwd.txt | grep home
usbmux:x:103:46:usbmux daemon,,,:/home/usbmux:/bin/false
saned:x:113:121::/home/saned:/bin/false
vol:x:1000:1000:vol,,,:/home/vol:/bin/bash
mark:x:1001:1001::/home/mark:/bin/bash
kali@kali:~/5ctf$
```

One other user that has a home directory not in home is root who has /bin/bash

Flag: mark,root,vol



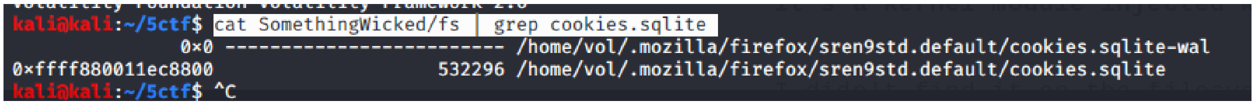
Something Wicked 5

100

Recover the cookies.sqlite of the vol user. What website stored the most cookies? Answer format should just be domain.TLD

First, we need to find the cookies.sqlite file with the following command that will dump the filesystem: volatility -f somethingwicked.bin --profile=Linuxprofilex64 linux_enumerate_files > SomethingWicked/fs

We can then run this command to locate the file: cat SomethingWicked/fs | grep cookies.sqlite



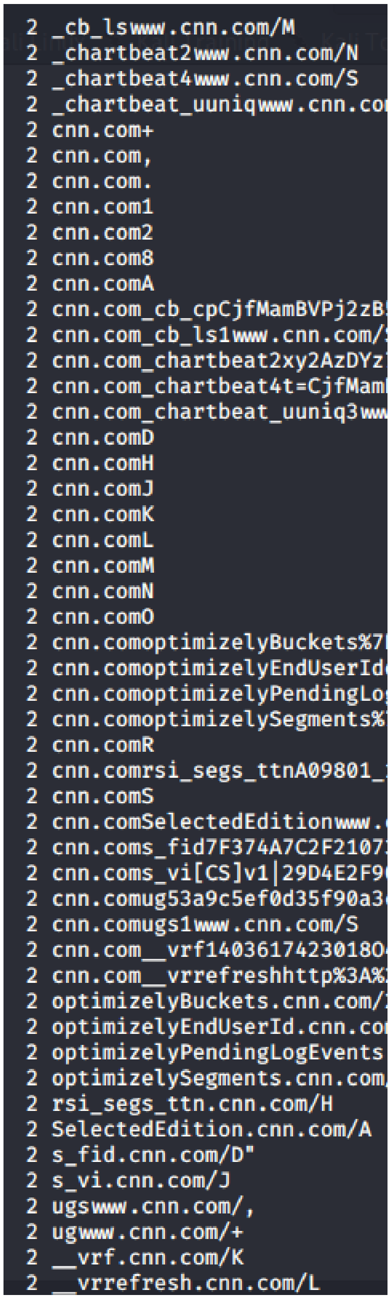
```
Volatility Foundation Volatility Framework 2.6
kali@kali:~/5ctf$ cat SomethingWicked/fs | grep cookies.sqlite
               0x0 ------------------------- /home/vol/.mozilla/firefox/sren9std.default/cookies.sqlite-wal
0xffff880011ec8800                     532296 /home/vol/.mozilla/firefox/sren9std.default/cookies.sqlite
kali@kali:~/5ctf$ ^C
```

So, the file we want is at: /home/vol/.mozilla/firefox/sren9std.default/cookies.sqlite

Let’s extract that file out: volatility -f somethingwicked.bin --profile=Linuxprofilex64 linux_find_file -i 0xffff880011ec8800 -O SomethingWicked/cookie.sqlite

We can run the following command to get some data out of the database: strings SomethingWicked/cookie.sqlite | sort | uniq -c

While it may not have the highest count, we see that there is one domain that is appearing in a bunch of the lines.



```
  2 _cb_lswww.cnn.com/M
  2 _chartbeat2www.cnn.com/N
  2 _chartbeat4www.cnn.com/S
  2 _chartbeat_uuniqwww.cnn.co
  2 cnn.com+
  2 cnn.com,
  2 cnn.com.
  2 cnn.com1
  2 cnn.com2
  2 cnn.com8
  2 cnn.comA
  2 cnn.com_cb_cpCjfMamBVPj2zB
  2 cnn.com_cb_ls1www.cnn.com/
  2 cnn.com_chartbeat2xy2AzDYz
  2 cnn.com_chartbeat4t=CjfMam
  2 cnn.com_chartbeat_uuniq3ww
  2 cnn.comD
  2 cnn.comH
  2 cnn.comJ
  2 cnn.comK
  2 cnn.comL
  2 cnn.comM
  2 cnn.comN
  2 cnn.comO
  2 cnn.comoptimizelyBuckets%7
  2 cnn.comoptimizelyEndUserId
  2 cnn.comoptimizelyPendingLo
  2 cnn.comoptimizelySegments%
  2 cnn.comR
  2 cnn.comrsi_segs_ttnA09801_
  2 cnn.comS
  2 cnn.comSelectedEditionwww.
  2 cnn.coms_fid7F374A7C2F2107
  2 cnn.coms_vi[CS]v1|29D4E2F9
  2 cnn.comug53a9c5ef0d35f90a3
  2 cnn.comugs1www.cnn.com/S
  2 cnn.com__vrf14036174230180
  2 cnn.com__vrrefreshhttp%3A%
  2 optimizelyBuckets.cnn.com/
  2 optimizelyEndUserId.cnn.co
  2 optimizelyPendingLogEvents
  2 optimizelySegments.cnn.com
  2 rsi_segs_ttn.cnn.com/H
  2 SelectedEdition.cnn.com/A
  2 s_fid.cnn.com/D"
  2 s_vi.cnn.com/J
  2 ugswww.cnn.com/,
  2 ugwww.cnn.com/+
  2 __vrf.cnn.com/K
  2 __vrrefresh.cnn.com/L
```

Flag: cnn.com