https://www.geeksforgeeks.org/awk-command-unixlinux-examples/?ref=leftbar-rightbar

https://www.journaldev.com/24871/awk-command-linux-unix

https://www.geeksforgeeks.org/awk-command-unixlinux-examples/?ref=leftbar-rightbar

# ********** Linux Basics **********

1. What command lists the contents of directories in Linux/Unix systems?
   - Ls
2. For the ls command, what arguments, or switch options, will allow you to print human-readable file sizes in a long-list format

   The flag is entire command, including argument

   ls –lh

3. What character will pipe the standard output from echo "I'm a plumber" to another command, as standard input?
   a. |
4. What argument/switch option, when used with man, will search the short descriptions and man-page-names for a keyword that you provide?The flag is the complete command, with argument/switch

   man -k

5. What is the absolute path to the root directory?
   a. /
6. What is the absolute path to the default location for configuration files?
   a. /etc

7. What is the directory that contains executable programs (binaries) which are needed in single user mode, to bring the system up or to repair it?
   a. /bin

8. What is the absolute path to the directory which contains non-essential binaries that are accessible by standard users as well as root
   a. /usr/bin

9. n absolute path to a directory which contains binaries only accessible by the root user, or users in the root group
   a. /sbin

10. What is the absolute path for the binary cat man-page?
    a. whereis cat… (cat0 will provide path )
    b. whereis -bm cat
        i. answ: /usr/share/man/man1/cat.1.gz

11. Search the man pages for the keyword *digest*. Then, use one of the binaries listed to hash the string **OneWayBestWay** using the largest *sha* hash available.
    a. To search fo the key word: man –k digest. Here k is keywaord
    b. To get the hash:echo –n your input | sha512sum (choose your hash algoritm)

Decryption: This file contains encrypted contents. Identify its file type, then decode its contents.

Ls – to see files
Dir – to see directories
File "name of file" - gives you the file type. For e,g file Encrypted is a zip file.
To unzip the file unzip (filename). Eg unzip Encrypted

To decrypt using SSL
openssl enc -d -aes-128-cbc -in cipher -out text.txt-----will prompt for the key
Here, enc is encryption

D is decrypt

-aes-12-cbc = is encryption type

-in = which file

-out = to which file

To read after cat filename

OR

man -k digest #displays algorithms, sha512sum is the highest

echo -n  'OneWayBestWay' | sha512sum

12. **Use File: /home/garviel/EncryptedThis file contains encrypted contents. Identify its file type, then decode its contents.**

    **Hint: Use OpenSSL**

    unzip Encrypted

    #there are 2 files in the zip named: cypher(encrypted text), symmetric(Key/password)

    openssl enc -d -aes256 -in cipher -out ans.txt

    #enter password from symmetric: AES128Key

cat ans.txt

DeCrypt

13. **Search the user home directories to find the file with the second-most lines in it. Hint: Exclude the VDI file! The flag is the number of lines in the file.**

- find ./ -type f | xargs wc -l

20000

14. **Read the file that contains the user database for the machine. Identify a strange comment.**

go to home dir

ls to see all the files

to find the length of the file in home dir

find /home -type f –exec wc –l {} \;

here, it means in home dir find file type in exec mode and give word count length

To sort them numerically, sort –n output

or

cat passwd | awk -F ':' '{print$5}'

Traitor

15. **Identify all members of the Lodge group. List their names in alphabetical order with a comma in between each name.**

a. To find the groups type command: groups
b. To list all member of group getent group (groupname)
    i. Getent group lodge(name)
    ii. To get the content of the group getent group(name)
    iii. e.g getent group lodge
    iv. To sort alphabetically (unknown)

cat group | grep "lodge"

#OR

cat group | grep 'lodge'| awk -F ':' '{print$4}'| awk -v RS='[,\n]' '{print$1}' |sort | tr '\n' ', '

aximand,erebus,ezekyle,garviel,sejanus,tarik

16. **Find the user with a unique login shell.**

- go to etc dir and cat the passwd file

cd /etc --------- cat passwd -----

e.g root:x:0:0:root:/root:/bin/bash (these are divided into 7 section, the last one is bash and 1st one is username)

- to sort through the file: awk –f ':' 'print{$1, $7}' /etc/passwd

nobody

OR

cat passwd | awk -F ":" '{print$1","$7}' | grep -wv "nologin"

returns mainly users with login shells that are bin/bash, therefore unique is the one that is different

#OR

cat passwd | awk -F ':' '{print$1$7}'

17. **Identify the algorithm, the amount of salted characters added, and the length of the hashed password in the file that stores passwords.**

**Hint: Research 'padding'...**

Flag format: *algorithm,#characters,#length*

*- /etc/shadow conatins he hashed password*

*- $6$ZMPKtXQ2$PoFwHOkiHYdAo/LQ/gPl6xjkIZ8f /yPxyWXaHPLFXOtnMuNykRsIspIg0MeHFx/gXwTcACeBFd3GotzAgACO50*

**$1$** is MD5

**$2a$** is Blowfish

**$2y$** is Blowfish

**$5$** is SHA-256

**$6$** is SHA-512

*To get the number of hashed password, count from $ZMPKtXQ2$ as this is the hashed used to the password: 8*

*To get the length of the password, count from PoFwHOkiHYdAo/LQ/gPl6xjkIZ8f /yPxyWXaHPLFXOtnMuNykRsIspIg0MeHFx/gXwTcACeBFd3GotzAgACO50*

*The rest of the numbers are when password expires etc*

*SHA 512,8,86*

18. **Find the directory named Bibliotheca. Enter the absolute path to the directory.**

sudo find "$(cd ..; pwd)" -name "Bibliotheca"

/media/Bibliotheca

OR

find / -type d -name "Bibliotheca”

OR

$ find / -name "Bibliotheca" -ls

19. **Identify the number of users with valid login shells, who can list the contents of the *Bibliotheca* directory.**

- to read directory getfacl /(directoryname)--- this gives full inofr of file, owner, group and who has what access to the directoery

To read who has access to the directory us: getent group (groupname)

To read file use: ls –lh /(directoryname)

Answ: 7

Do ls -lisa on directory to find who has permissions and what group. Since all have permissions to execute just find all users with valid login shells using:

awk -F: 'NR == FNR {shells[$0]; next} $NF in shells' /etc/{shells,passwd} /media$ ls -ld Bibliotheca

20. **The permissions that user sejanus has on */media/Bibliotheca*, in octal format.**

**Flag format: #**

**HINT: Think about groups...**

**Here, from above information we know the group is chapter and seajnus falls under that group.**

**Chapter has read and execute access (r-read, x-execute)**

- **r**(ead) has the value of **4**
- **w**(rite) has the value of **2**
- (e)**x**(ecute) has the value of **1**
- **no permission** has the value of **0**

So adding r+x = 5

cd /media

Ls -lisa #see permissions (rwxr-xr-x) rxr is user, r-x is group, r-x is other

R-x is 5 in octal for the group Chapter

Senjanus is in the group Chapter (5 is answer)

21. **Locate the file within /media/Bibliotheca that is modifiable by the only user that is part of the Chapter group, but not part of the Lodge group.**

Hint: Not the hidden file...

To see the files in directory ls \(Directoryname)---

To read the files with uses, group and access listed: ls -lh Bibliotheca/Bibliotheca_unus/

Here, w is execute access so asnweis codex_Astartes

cat group | grep "chapter"

ls -R -la #long Version

ls -R -la | tr ' ' '' |*awk -F* '' '{print$1$3$4$10}' |grep mephiston

#Cleaned up notice user writes RW for mephiston and not group or other

Codex_Astartes

22. **Identify the file within */media/Bibliotheca* where the owning group has more rights than the owning user.**

To list the files ls –lh(file name with full path ) e.g ls –lh Bibliotheca/Bibliotheca_tribus/

Codex_Hereticusls

ls -R -la | tr ' ' '' |*awk -F* '' '{print$1$3$4$10}' #notice r for user but rw for group Codex_Imperium

- r--rw-r-- **group has more permissions than owner

23. **Identify the file within */media/Bibliotheca* where the owning group has more rights than the owning user.**
First check if the file is executable usinf ls –l (file name).

E.g: ls -l Bibliotheca/Bibliotheca_tribus/Codex_Imperium

Here, no execute access. To change the access

24. **The user tyborc is unable to access the directory:**

    **/media/Bibliotheca/Bibliotheca_unus**

    **Why? Identify the permission missing in standard verb form.**

    Cd /media/Bibliotheca

    cat group | grep tyborc # member of guardmen group

    ls -R -la | grep unus #shows mephison and Chapter so tyborc would be other

    dr-xr-xr-- 2 mephiston chapter  4096 Feb 28 19:05 Bibliotheca_unus # other is r–

    Missing execute

25. **Locate the file in /media/Bibliotheca that Inquisitor Quixos has sole modification rights on.**

    **The flag is the absolute path for the file.**

    ls -R -la | grep quixos  #in media bibliotheca

    RETURNS 4 RESULTS:

    rw------- 1 quixos quixos 3609 Feb 28 19:05 Codex_Hereticus

    - --xr-xrwx 1 quixos quixos 3609 Feb 28 19:05 Codex_Hereticus
    - r----xrwx 1 quixos quixos 3609 Feb 28 19:05 Codex_Hereticus
    - r--r--r-- 1 quixos quixos 3609 Feb 28 19:05 Codex_Hereticus

BACK OUT TO /

find -name 'Codex_Hereticus'

RETURNS 4 PATHS

./media/Bibliotheca/Bibliotheca_unus/Codex_Hereticus

./media/Bibliotheca/Bibliotheca_duo/Codex_Hereticus

./media/Bibliotheca/Bibliotheca_quattuor/Codex_Hereticus

./media/Bibliotheca/Bibliotheca_tribus/Codex_Hereticus

cd Bibliotheca_duo #DOUBLE CHECKING THIS IS THE CORRECT ONE

ls -lisa |grep quixos

258835  4 -rw------- 1 quixos  quixos  3609 Feb 28 19:05 Codex_Hereticus

So, “./media/Bibliotheca/Bibliotheca_duo/Codex_Hereticus” is correct

**26. Read the concealed file within /media/Bibliotheca**

i

cd /media/Bibliotheca/Bibliotheca_duo/

ls -Al

cat .Secrets_of_the_Immaterium

Expand your mind

**27. Find the warp and read its secrets for the flag.**

find -mindepth 1 -name '.*' -type f

cd Bibliotheca_duo/.warp2/.warp5/warp5/.warp3/warp2/.secrets

ls -Al

cat .secrets

Ph'nglui mglw'nafh Cthulhu#OR

find -name "." *#RETURNS 3 ODD NAMES ##find -mindepth 1 -name '*.' -type f is better

cd "Bibliotheca_duo/.warp2/.warp5/warp5/.warp3/warp2"

cat .secrets

Ph'nglui mglw'nafh Cthulhu

28. Execute the file owned by the **guardsmen** group in */media/Bibliotheca*, as the owning user.

The flag is the code name provided after a successful access attempt.

To check what access you have do sudo –l ----- this provides information on what users can you run file as

Now find the file owned by guardsmen and go to that directory to execute the file:

To run the file sudo –u gaunt(username) ./(filename)

29. **The user tyborc is unable to access the directory: /media/Bibliotheca/Bibliotheca_unus**

**Why? Identify the permission missing in standard verb form.**

- Go inside the directory cd/media

- **cat** stands for concatenate to create single or multiple files, view content of a file, concatenate f iles and redirect output in terminal or files

- sorting multiple files in a single file

Cd /media/Bibliotheca

cat group | grep tyborc # member of guardmen group

ls -R -la | grep unus #shows mephison and Chapter so tyborc would be other

dr-xr-xr-- 2 mephiston chapter 4096 Feb 28 19:05 Bibliotheca_unus # other is r–

Missing execute

30. ***You only have a single submission attempt for this challenge.***

Locate the file in */media/Bibliotheca* that **Inquisitor Quixos** has sole modification rights on.

The flag is the absolute path for the file.

ls -R -la | grep quixos #in media bibliotheca# -la shows the hidden file and using grep it will l ook for quixos through the directory

BACK OUT TO /

find -name 'Codex_Hereticus'

RETURNS 4 PATHS

cd Bibliotheca_duo #DOUBLE CHECKING THIS IS THE CORRECT ONE

ls -l|grep quixos #shows the file in ling format with all the information

Here, this group has the sole permission to edit the file

“./media/Bibliotheca/Bibliotheca_duo/Codex_Hereticus” is correct

31. **Read the concealed file within *`/media/Bibliotheca`***

catcd /media/Bibliotheca/Bibliotheca_duo/

ls –Al

Here, "." in front of the file means it is a hidden file

cat .Secrets_of_the_Immaterium

32. **Find the warp and read its secrets for the flag.**

From previous question we see warp files

To search for the files you can use the following command combined with find

find . -exec grep -Hn warp {} \;

Or

Use grep to find the exact file with case senstive use the command:

grep –ri –l "word"

To search for the file secrets, use the command

Ls –aR | grep –b5 secret

Here, ls(look) for –ar(Hidden files recursively) and grep folders before the file(secret)

Now we know the folder path, we can use cat to read the file:

cat /media/Bibliotheca/Bibliotheca_duo/.warp2/.warp5/warp5/.warp3/warp2/.secrets

33. **Using the commands ls and grep, identify the number of directories in /etc/ that end in .d**

ls -al | grep '\.d$' | wc -l

Here, .d stands for directory name

Wc = word count

34. **File: home/garviel/numbers**

    **Use regular expressions to match valid IP addresses. The flag is the number of addresses.**

    **HINT**: What are the valid numerical values of each octet in an IP address?

--- read the numbers file first and then grep it:

Cat numbers | grep –E ‘'\b((25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?)(\.|$)){4}\b' # this is the IP format, the output will be following:

To count the files Cat numbers | grep –E ‘'\b((25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?)(\.|$)){4}\b' | wc –l

18

35. **File: home/garviel/numbers**

    **Use regular expressions to match patterns that look similar to a MAC Address. Flag is a count of the number of matches.**

    **HINT: This is a loose match! Some of these results won't be true MAC addresses.**

- here to look for the MAC using the following expression:

cat numbers | grep -E '^([0-9A-Za-z]{2}[:-]){5}([0-9A-Za-z]{2})$'

Here, 0-9 A-Za-z == means look for all the alphabets including CAP’s and small and digits 0-9

{2} = means do the same twice

{5} = do it 5 times

cat numbers | grep -E '^([0-9A-Za-z]{2}[:-]){5}([0-9A-Za-z]{2})$' | wc –l #to count the value

- 4877

36. **File: home/garviel/numbers**

    **Use awk to print lines:**

>= 420 AND <=1337

The flag is a SHA512 hash of the output.

Awk = used for compiling and generating reports by processing files

NR built in variable stands for number of records

awk 'NR >= 420 && NR <= 1337' numbers | sha512sum

37. Directory: home/garviel/Battlefield/

    The garviel user has a minefield map and controls to a Titan War Machine located in their home directory. Interpret the Titan Controls to navigate the minefield and annihilate the target.

    Enter the correct movement codes to make the Titan obliterate the target.

    AAAAA3AAA3AAAABAABAAAA

38. **The flag resides in $HOME/paths... you just need to determine which flag it is. The flag sits next to a string matching the name of a $PATH/binary on your system.**

    Hint: The correct binary is not echo

    Binaries Path--- usr/bin  and usr/sbin

    Here,

    Go to home/garviel and open the file

    Go to usr binary in this case usr/bin  ---- send the output of this file to:

    sudo ls > /home/garviel/output.txt

    Once both outputs are there compare the paths and output.txt file for a common string

    grep -Ff /home/garviel/output.txt paths

    Here, we are comparing output.txt with path. Once we get complete string there are 2  of them.. Python and viewless

    To see which one os on the output.txt file : cat output.txt | grep python

    - this gives the answer

39. **File: home/garviel/connections**

    **Use awk to create a separate CSV (comma separated value) file that contains columns 1-6.**

    **The flag is an MD5 hash of the new file**

    **Hint: Look at #fields on line 6 in the file to understand column layout.**

    **Hint: This is a Zeek (formally known as Bro) connection log file in TSV format. Click This Link to learn about its formatting.**

    awk -F '\x09' '{print$1","$2","$3","$4","$5","$6}' <connections>output.csv

    Here, read the connection file separated by \x09 and print all 6 line separated by comma and output to output.csv

To get the md5sum md5su output.csv

**33. To complete this challenge, find the description of the Lego Land service.**

Get-WmiObject -Class Win32_Service -Filter "name='legoland'" | ft description

i_love_legos

**The flag resides in $HOME/paths... you just need to determine which flag it is. The flag sits next to a string matching the name of a $PATH/binary on your system.**

**Hint: The correct binary is not echo**

cat paths | awk -F ' ' '{print$1}' > new1.txt

SEE bin paths using $PATH #LS -AL all paths to find directories and copy them to files

/usr/local/sbin —-> NO DIRECT NAMES

/usr/local/bin —-> No DIRECT NAMES

/usr/sbin —->  ls -al | awk -F ' ' '{print$9}' > /home/garviel/usrsbin.txt —> NO HITS

/usr/bin —-> ls -al | awk -F ' ' '{print$9}' > /home/garviel/usrbin.txt —> Python3 { Vrc0vw7ZUaLBpQp  }

/sbin —->  ls -al | awk -F ' ' '{print$9}' > /home/garviel/sbin.txt

/bin  —->  ls -al | awk -F ' ' '{print$9}' > /home/garviel/bin.txt

/usr/games —>  ls -al | awk -F ' ' '{print$9}' > /home/garviel/usrgames.txt

/usr/local/games —--> NO DIRECT NAMES

/snap/bin —------> NO DIRECT NAMES

cat new1.txt X.txt |sort|uniq -c | grep 2 #compare both files for all

#OR

$PATH #this will show you the bash of all the binary files

cat paths # this will show you all of the possible flags

ls /usr/sbin > blah.txt #copy contents of binary into text file

ls /usr/bin >> blah.txt #append the answer

ls /bin >> blah.txt

ls /sbin >> blah.txt

cat blah.txt

cat paths | grep -f blah.txt -F #match the words that are the same between paths and $PATH

python3 is the word that has a complete match when you grep

Vrc0vw7ZUaLBpQp

## ********** Linux Boot Process **********

1. **Solve the following equation:**

   **0x31A - 0x21B**

   **Enter the flag in Hexadecimal form.**

   Calculate each value in decimal and subtract the sum: 255. Convet 255 to hex 0xFF

2. **How many bits are in a nibble, and a byte?**
   a. 4,8

3. **How many bits does a single Hexadecimal character represent?**
   a. 4

4. **Each hex digit contains a value of 8 bits when used to represent memory. How many bytes could the range 0x00000000 - 0x00000010 contain?**
   a. 0x00000000 = 0
   b. 0x000000010 = 16

      so from 0 to 16 is “17”

5. **How large is the Master Boot Record and what directory is it located in?**

   Flag format: #InBytes, director

   - 512, /dev
   -

6. **Identify which of your Linux machines is using SysV Initialization.**
   a. Minas-Tirith : Answer is in the flag

7. **What are the maximum and minimum value a single Hexadecimal digit can contain?**

**Enter the values in Linux Hexadecimal Numerical Constant form.**

**Flag format: min-max**

Min – 0 Max-15

Answ: 0x0 – 0xF

8. **Solve the following equation:0x31A + 0x43. Enter the flag in Hexadecimal form.**
   a. 0x31A = 794, 0x43 = 5 === 861==0x35D

9. **Execute : sudo cat /dev/vda | xxd -l 32 -c 0x10 -g 1**

   **What are the values contained in hex positions 0x00000001 through 0x00000008?**

   **Flag format: Value,Value,Value**

```
bombadil@minas-tirith:/$ sudo cat /dev/vda | xxd -l 32 -c 0x10 -g 1
00000000: eb 63 90 8e d0 31 e4 8e d8 8e c0 be 00 7c bf 00  .c...1.......|..
00000010: 06 b9 00 01 f3 a5 be ee 07 b0 08 ea 20 06 00 00  ............ ...
```

10. **Locate the master boot record for one of the Linux machines and read it with xxd**

    **What programming language is the MBR written in?HINT: Look at the first three bytes**

    - Assembly

11. **The file /home/bombadil/mbroken is a copy of an MBR from another machine.**

    **Hash the first partition of the file using md5sum. The flag is the hash.**

12. **The file /home/bombadil/mbroken is a copy of an MBR from another machine.**

    **You will find the "word" GRUB in the output, hash using md5sum.**

    **The flag is the entire hash.**

13. **The file /home/bombadil/mbroken is a copy of an MBR from another machine.**

    **Hash only the Bootstrap section of the MBR using md5sum. The flag is the entire hash.**

14. **Identity the default run level on the SysV Init Linux machine.**

**Flag format: #**

cat /etc/inittab

#or

who –r

who command is a tool print information about users who are currently logged in

15. **What is the last script to run when the command init 6 is executed?**

    **Flag format: /absolute/path**

    **NOTE: Use the machine identified in SysV 1 for this question.**

    ls -l /etc/rc6.d  # rc6. d is for **reboot**

    /etc/init.d/reboot (answ)

```
bombadil@minas-tirith:~$ ls -l /etc/rc6.d
total 4
lrwxrwxrwx 1 root root  19 Oct 13  2021 K01cgmanager -> ../init.d/cgmanager
lrwxrwxrwx 1 root root  17 Oct 13  2021 K01cgproxy -> ../init.d/cgproxy
lrwxrwxrwx 1 root root  22 Feb  9  2020 K01cloud-config -> ../init.d/cloud-config
lrwxrwxrwx 1 root root  21 Feb  9  2020 K01cloud-final -> ../init.d/cloud-final
lrwxrwxrwx 1 root root  20 Feb  9  2020 K01cloud-init -> ../init.d/cloud-init
lrwxrwxrwx 1 root root  26 Feb  9  2020 K01cloud-init-local -> ../init.d/cloud-init-local
lrwxrwxrwx 1 root root  20 Feb  9  2020 K01irqbalance -> ../init.d/irqbalance
lrwxrwxrwx 1 root root  15 Feb  9  2020 K01unscd -> ../init.d/unscd
lrwxrwxrwx 1 root root  17 Oct 13  2021 K01urandom -> ../init.d/urandom
lrwxrwxrwx 1 root root  18 Oct 13  2021 K02sendsigs -> ../init.d/sendsigs
lrwxrwxrwx 1 root root  17 Oct 13  2021 K03rsyslog -> ../init.d/rsyslog
lrwxrwxrwx 1 root root  20 Oct 13  2021 K04hwclock.sh -> ../init.d/hwclock.sh
lrwxrwxrwx 1 root root  22 Oct 13  2021 K04umountnfs.sh -> ../init.d/umountnfs.sh
lrwxrwxrwx 1 root root  20 Oct 13  2021 K05networking -> ../init.d/networking
lrwxrwxrwx 1 root root  18 Oct 13  2021 K06umountfs -> ../init.d/umountfs
lrwxrwxrwx 1 root root  20 Oct 13  2021 K07umountroot -> ../init.d/umountroot
lrwxrwxrwx 1 root root  16 Oct 13  2021 K08reboot -> ../init.d/reboot
-rw-r--r-- 1 root root 351 Feb 12  2017 README
```

16. **What unit does the graphical.target want to start, based solely on its configuration file?**

    **HINT: Targets deal with which init system? Which machine should you be looking for this flag, on?  NOTE: Use the SystemD Machine for this question.**

17. **What dependency to graphical.target will stop it from executing if it fails to start, based solely on its static configuration file?**

18. **How many wants dependencies does SystemD actually recognize for the default.target**

    **HINT: Use the systemctl command with some arguments to make life easier.**

    **Flag format: #**

19. **What is the full path to the binary used for standard message logging?**

    **HINT: Standard message logging is standardized across UNIX systems.**

    **NOTE: As the challenge name suggests, use the SystemD machine for this question.**

    **Flag format: /absolute/path**

20. **Identify the Linux Kernel being loaded by the Grub, by examining its configuration.**

    **Enter the command used by the Grub, and the full path to the Kernel, as the flag.**

    **Flag Format: command,kernel location**

    **HINT:**

    **Click me for help understanding Grub commands**

    **Machine: Minas_Tirith**

## ********** Linux Process Validity **********

**1. What is the process ID (PID) of the SysV Init daemon?**

1

https://uace.github.io/learning/init-vs-systemd-what-is-an-init-daemon

**2. How many child processes did SysV Init daemon spawn?**

**ps** displays information about a selection of the active processes

Here, we know init has a PID of 1. To see child processes of this main process use:

pgrep -P $(your_process1_pid)

To grab the length

bombadil@minas-tirith:/$ pgrep -P 1 | wc -l

```
bombadil@minas-tirith:/$ pgrep -P 1
369
1074
1109
1252
1281
1284
1349
1414
1444
1469
1705
1775
1776
1777
1778
1779
1780
1781
1782
30667
bombadil@minas-tirith:/$ pgrep -P 1 | wc -l
24
```

https://stackoverflow.com/questions/17743879/how-to-get-child-process-from-parent-process

**3. Identify all of the arguments given to the ntpd daemon (service) using ps.**

Here, Elf produce the most complete information for each process

Now using ps to grep the ntpd we ger following results:

ps -elf | grep -I “ntpd”

-p /var/run/ntpd.pid -g -u 105:109

```
ombadil@minas-tirith:/$ ps -elf | grep -i "ntpd"
 S ntp       1414     1  0  80   0 - 23949 -      Jul07 ?       00:01:05 /usr/sbin/ntpd -p /var/run/ntpd.pid -g -u 105:109
 S bombadil  8545  3009  0  80   0 -  3179 -      16:25 pts/0    00:00:00 grep -i ntpd
```

https://www.xitalogy.com/linux-unix/2020/02/22/ps-elf-filter-results-effectively-and-see-if-a-linux-process-is-running.html

**4. What is the parent process to Bombadil’s Bash process?**

- use command ps –efj to see all the running process

- to find the bash use: bombadil@minas-tirith:/$ ps -efj | grep "bash" ----- this will give us 3 results

Now to get the parent pocess grep the PPID (PPID is parent process to PID), PPID is 3rd column

ps -efj | grep 3045

#Note even if I grep 3006 it still gives the **sshd** as the parent process

```
bombadil@minas-tirith:/$ ps -efj | grep "bash"
bombadil  3009  3006  3009  3009  0 15:07 pts/0    00:00:00 -bash
bombadil  3046  3045  3046  3046  0 15:07 pts/1    00:00:00 -bash
bombadil  9893  3009  9892  3009  0 16:44 pts/0    00:00:00 grep bash
bombadil@minas-tirith:/$ ps -efj | grep 3045
bombadil  3045  3031  3031  3031  0 15:07 ?        00:00:00 sshd: bombadil@pts/1
bombadil  3046  3045  3046  3046  0 15:07 pts/1    00:00:00 -bash
bombadil  9955  3009  9954  3009  0 16:45 pts/0    00:00:00 grep 3045
```

**5. Identify the file mapped to the fourth file descriptor (handle) of the cron process.**

**HINT: There might be multiple cron processes, but only one with the answer.**

**Flag format: /absolute/path**

Lsof = list of open files command

Sudo Lsof –c  (your process name) # in this case cron to view the list of each file opened by a particular process ID

Now, to read the files find a number starting with 0---- we count this with 1 so the 4th file is

/run/crond.pid

```
cron    1349 root  mem    REG             254,1      368    24007 /usr/lib/locale/en_US.utf8/LC_IDENTIFICATION
cron    1349 root    0r   CHR               1,3      0t0     1028 /dev/null
cron    1349 root    1w   CHR               1,3      0t0     1028 /dev/null
cron    1349 root    2w   CHR               1,3      0t0     1028 /dev/null
cron    1349 root    3u   REG              0,19        5    12320 /run/crond.pid
cron   12840 root  cwd    DIR             254,1     4096    16636 /var/spool/cron
```

https://www.thegeekstuff.com/2012/08/lsof-command-examples/

**6. Identify the permissions that cron has on the file identified in Processes 5.**

**HINT: Read the man page for lsof to understand permissions.**

**Flag format: If more than one, list the permissions comma separated, no spaces**

Sudo lsof –c “cron” # to View the files-----when we scroll down, we can see 5 for processes

```
bombadil@minas-tirith:/$ sudo lsof -c "cron"
COMMAND  PID USER   FD   TYPE DEVICE SIZE/OFF  NODE NAME
cron    1349 root  cwd    DIR  254,1     4096 16636 /var/spool/cron
cron    1349 root  rtd    DIR  254,1     4096     2 /
cron    1349 root  txt    REG  254,1    48616 10658 /usr/sbin/cron
cron    1349 root  mem    REG  254,1   135440  3104 /lib/x86_64-linux-gnu/libpthread-2.24.so
cron    1349 root  mem    REG  254,1    22944  1944 /lib/x86_64-linux-gnu/libcap-ng.so.0.0.0
cron    1349 root  mem    REG  254,1   468920  2450 /lib/x86_64-linux-gnu/libpcre.so.3.13.3
cron    1349 root  mem    REG  254,1    14640  1577 /lib/x86_64-linux-gnu/libdl-2.24.so
cron    1349 root  mem    REG  254,1   120752   940 /lib/x86_64-linux-gnu/libaudit.so.1.0.0
cron    1349 root  mem    REG  254,1  1689360  1574 /lib/x86_64-linux-gnu/libc-2.24.so
cron    1349 root  mem    REG  254,1   155400  2012 /lib/x86_64-linux-gnu/libselinux.so.1
cron    1349 root  mem    REG  254,1    56016  2411 /lib/x86_64-linux-gnu/libpam.so.0.83.1
cron    1349 root  mem    REG  254,1   153288  1569 /lib/x86_64-linux-gnu/ld-2.24.so
cron    1349 root  mem    REG  254,1   328180 21947 /usr/lib/locale/aa_DJ.utf8/LC_CTYPE
cron    1349 root  mem    REG  254,1       54 22033 /usr/lib/locale/aa_ET/LC_NUMERIC
cron    1349 root  mem    REG  254,1     2454 24008 /usr/lib/locale/en_US.utf8/LC_TIME
cron    1349 root  mem    REG  254,1  1244054 21946 /usr/lib/locale/aa_DJ.utf8/LC_COLLATE
cron    1349 root  mem    REG  254,1      286 23637 /usr/lib/locale/chr_US/LC_MONETARY
cron    1349 root  mem    REG  254,1       57 23853 /usr/lib/locale/en_AG/LC_MESSAGES/SYS_LC_MESSAGES
cron    1349 root  mem    REG  254,1       34 23639 /usr/lib/locale/chr_US/LC_PAPER
cron    1349 root  mem    REG  254,1    26258  6396 /usr/lib/x86_64-linux-gnu/gconv/gconv-modules.cache
cron    1349 root  mem    REG  254,1       77 23638 /usr/lib/locale/chr_US/LC_NAME
cron    1349 root  mem    REG  254,1      167 24006 /usr/lib/locale/en_US.utf8/LC_ADDRESS
cron    1349 root  mem    REG  254,1       59 23640 /usr/lib/locale/chr_US/LC_TELEPHONE
cron    1349 root  mem    REG  254,1       23 23635 /usr/lib/locale/chr_US/LC_MEASUREMENT
cron    1349 root  mem    REG  254,1      368 24007 /usr/lib/locale/en_US.utf8/LC_IDENTIFICATION
cron    1349 root    0r   CHR    1,3      0t0  1028 /dev/null
cron    1349 root    1w   CHR    1,3      0t0  1028 /dev/null
cron    1349 root    2w   CHR    1,3      0t0  1028 /dev/null
cron    1349 root    3u   REG   0,19        5 12320 /run/crond.pid
```

In file descriptor it has u, which stands for read and write permission

# read,write

7. **Identify the names of the orphan processes on the SysV system.**

Htop

Look at the last column it will pop up with orphan processess

```
18215  1781 S root      20   0  4028   648   584 S  0.0  0.0  0:00.00   └─ /home/orphan
18219 18215 S root      20   0  4028    72     0 S  0.0  0.0  0:00.00      ├─ BruceWayne_
18218 18215 S root      20   0  4028    72     0 S  0.0  0.0  0:00.00      ├─ Aragorn____
18217 18215 S root      20   0  4028    72     0 S  0.0  0.0  0:00.00      ├─ Eowyn______
18216 18215 S root      20   0  4028    72     0 S  0.0  0.0  0:00.00      └─ Tolkien___Main
 1780     1 S root      20   0  4440   680   616 S  0.0  0.0  0:00.00 ├─ /sbin/getty 38400 tty6
 1779     1 S root      20   0  4440   688   628 S  0.0  0.0  0:00.00 ├─ /sbin/getty 38400 tty5
```

Aragorn,BruceWayne,Eowyn,Tolkien

**8. Locate zombie processes on the SysV system.**

**Identify the zombie processes' parent process.**

/bin/funk

**9. Locate the strange open port on the SysV system.**

**Identify the command line executable and its arguments.**

**Flag format: /executable/path -arguments**

sudo lsof -i -P -n | grep LISTEN  #this shows the listening port 9999 seems suspicious

Now reading all the processes and looking for 99999

Ps –elf #to look for running process

/bin/netcat -lp 9999

**10. Examine the process list to find the ssh process. Then, identify the symbolic link to the absolute path for its executable in the /proc directory.**

**The flag is the absolute path to the symbolic link, and the file it is linked to.**

**Flag format: /absolute/path,/absolute/path**

Ps –elf | grep ssh  ##Look for ssh process using the command:

/usr/sbin/sshd  ##This gives us the first absolute path

Now, to check for the second path using grep and following PID 16885 of bombadil@pts/0

Sudo ls –l /proc/1688 #this gives us the exe

exe -> /usr/sbin/sshd

/usr/sbin/sshd, /proc/1688/exe

11. I**dentify the file that contains udp connection information. Identify the process using port 123.**

**For the flag, enter:**

**Process name, File descriptor number for the udp socket**

**Its permissions as shown in lsof**

**Flag format: name,#,permission**

sudo lsof -i:123 #return all services using port 123

#The problem asks for a file that contains a UDP connection, so you must look for an IP address. Only one of these results has an IP address

ntp,19,u

```
bombadil@minas-tirith:/$ sudo lsof -i:123
COMMAND  PID USER   FD   TYPE DEVICE SIZE/OFF NODE NAME
ntpd    1414  ntp   16u  IPv6  11475      0t0  UDP *:ntp
ntpd    1414  ntp   17u  IPv4  11478      0t0  UDP *:ntp
ntpd    1414  ntp   18u  IPv4  11483      0t0  UDP localhost:ntp
ntpd    1414  ntp   19u  IPv4  11485      0t0  UDP 10.5.0.7:ntp
ntpd    1414  ntp   20u  IPv6  11487      0t0  UDP ip6-localhost:ntp
ntpd    1414  ntp   21u  IPv6  11489      0t0  UDP [fe80::f816:3eff:fe26:e12c]:ntp
```

**12. Locate the strange open port on the SysV system. Identify how the process persists**
**b** **etween reboots. The flag is the absolute path for the file that contains the persistence mechanism, and the configuration option.**

**HINT: Persistence is defined here**

**Flag format: filepath,configuration_option**

```
  PID  PPID S USER     PRI  NI  VIRT   RES   SHR S CPU% MEM%   TIME+  Command
    1     0 S root      20   0 15812  1996  1828 S  0.0  0.1  0:11.40 init [2]
30667     1 S root      20   0 37984  4332  3860 S  0.0  0.2  0:00.01 ├─ /lib/systemd/systemd-logind
 1782     1 S root      20   0  6300  1668  1560 S  0.0  0.1  0:00.00 ├─ /bin/netcat -lp 9999
 1781     1 S root      20   0  4028   912   840 S  0.0  0.0  0:02.56 ├─ /sauron
```

htop #will show you that port 9999 is the strange open port, so netcat is the persistence mechanism

cd /etc  #to change the directory to etc and to search for netcat

grep -R "netcat"  #look at bottom

/etc/inittab,91:2345:respawn:/bin/netcat -lp 9999

**13. Identify one of the human-readable file handles by the other program that creates a zombie process.**

**NOTE: Remember, zombie processes only live until the parent process kills them.**
**Try monitoring the processes list with top or htop to find them.**

**The flag is the text from one of the files it reads.**

**14. Scenario: The Villains group has been chanting offerings to their new leader at regular intervals over a TCP connection.**

**Task: Identify their method of communication and how it is occurring. Locate the following artifacts: ** The chant/text used by each villain (include spaces) ** The new Lord receiving the offering ** The IP address and port that the offering is received over**

**Flag format: *chant text,new Lord,IP:port***

**Machine: Minas_Tirith**

**15. Scenario: Someone or something is stealing files with a .txt extension from user directories. Determine how these thefts are occurring. Task: Identify the command being ran and how it occurs**

**Flag format: command,how it occurs**

**16. Scenario: Text files are being exfiltrated from the machine using a network connection. The connections still occur post-reboot, according to network analysts.**

**The junior analysts are having a hard time with attribution because no strange programs or ports are running, and the connection seems to only occur in 60-second intervals, every 15 minutes.**

**Task: Determine the means of persistence used by the program, and the port used. The flag is the command that allows exfiltration, and the file its persistence mechanism uses.**

**Flag format: command,persistence**

**17. Scenario: The web server has been modified by an unknown hacktivist group. Users accessing the web server are reporting crashes and insane disk usage.**

**Task: Identify the Cyber Attack Method used by the group, and the command running. Flag format: method,command**

**18. Scenario: Analysts have found a dump of commands on the Internet that refer to the Terra machine. The command history for one of the users with an interactive login is being stolen via unknown means. The network analysts can't find any persistent connections, but notice a spike in traffic on logon and logoff. Task: Identify how the command history is stolen from the machine.**

**The flag is the file used to execute the commands, and where they are sent. Flag format: /absolute/path/to/file,IP:port**

# ******** Linux Auditing and Logging **********

**1. File: /home/garviel/output.xml**

**Identify the XML element name in the output below**

**<scaninfo type="syn" protocol="tcp" numservices="200" services="1-200"/>**

https://www.w3schools.com/xml/xml_elements.asp

**2. Identify one of the XML attributes in the output below**

**<scaninfo type="syn" protocol="tcp" numservices="200" services="1-200"/>**

type=”syn” (answ)

**3. What RFC is Syslog?**

RFC 5424

**4. What is the numerical code assigned to the facility dealing with authorization?**

- 4

**5. How many severity codes are defined in the standard that defines syslog?**

- 8

**6. What severity is assigned to system instability messages?**

- 0

**7. In the legacy rules section of the file, what facility is logged to 0.log?**

| Selectors | Action |
|---|---|
| facility. severity | /path/to/log/location |

Ans: 0

**8. In the legacy rules section of the file, how many severities are logged to 0.log?**

- 8 severities as it includes all

**9. List the severities from highest severity (lowest numerical listed) to lowest severity (highest numerical listed) using their severity name.**

emergency,alert,critical,error,warning

**10. In the legacy rules section of the file, how many severities are logged to 4sig.log?**

List the severities from highest severity (lowest numerical listed) to lowest severity (highest numerical listed), using their severity name.

Prefixed with an exclamation point (!), it indicates the opposite, in other words the strictly lower priorities. So anything from 4 and below is not included.

Notice,informational,debug

https://debian-handbook.info/browse/vi-VN/stable/sect.syslog.html

**11. What is being logged in not.log?**

Provide the facilities from lowest facility to highest facility numerically, and the severity being logged. (List only the first word for each.)

mail,clock,NTP,notice

**12. What facilities and what severities are being sent to a remote server over a reliable connection using port 514?**

- Auth,authpriv,8,10.30.0.1

13. **Do logs that match this filter ever get saved on the local machine?**

Yes

**14. What messages are being sent to 10.84.0.1?**

**Provide the facility number, the number of severity codes, and Layer 4 connection type as the answer.**

**Flag format: F,S,Layer 4 connection Type**

0,7,udp

15**. File: /home/garviel/output.xml**

**Parse all of the IP addresses from the file using XPATH queries**

To parse the IP address use the following command:

xpath -q -e '//element/@attribute' file.xml

In our case: xpath -q -e '//address/@addr' output.xml

To get the md5 hash

xpath -q -e '//address/@addr' output.xml | md5sum

Ans: 0e850f14fc192c5105955ec094287bd2

**16. File: /home/garviel/output.xml**

**Select all of the IP addresses and ports using a single XPATH Union Statement**

To combine two information we use "|"

xpath -q -e '//address/@addr | //port/@portid' output.xml

Now, to get the hash: xpath -q -e '//address/@addr | //port/@portid' output.xml | md5sum

Ff7990139b6d09aa65afb6e069db0dec

http://www.tizag.com/xmlTutorial/xpathbar.php

**17. File: /home/garviel/conn.log**

**Use jq to pretty print the JSON file conn.log.**

**Hash the pretty-printed file with md5sum for the flag.**

To read the jason file:

jq . File.json, In our case: jq . Conn.log

To get md5 jq . Conn.log | md5sum

25ebedf7442e470eaaa48b5f7d5b96f4

**18. File : /home/garviel/conn.log This file is a conn.log made in Zeek (Bro) with data about TCP/IP connections.**

**Use jq to locate and count the unique originating endpoint IP addresses in the file. Enter the number of unique originating IP addresses as the flag.**

d.orig_h addr Originating endpoint's IP address

Here, in JSON file anything after . Is being read. So, for this example we have:

jq '."id.orig_h"' conn.log = This reads just the id.orig_h from thefile

Now to sort the file for unique IP address:

jq '."id.orig_h"' conn.log | sort | uniq

Now, to count we use wc –l

```
garviel@terra:~$ jq '."id.orig_h"' conn.log | sort | uniq
"10.50.20.196"
"10.50.20.87"
"10.50.22.97"
"10.50.23.205"
"10.50.23.242"
"10.50.23.62"
```

```
garviel@terra:~$ jq '."id.orig_h"' conn.log | sort | uniq | wc -l
31
```

**19. File: /home/garviel/conn.log This file is a conn.log made in Zeek (Bro) with data about TCP/IP connections.**

**Use jq to locate and count connections where the destination IP sent more than 40 bytes to the source IP.**

Destination IP = id.resp_h addr Responding endpoint's IP address (AKA RESP)

Destination ip bytes, the element we are looking for is resp_ip_bytes.

To get the output of thr response bytes: jq '."resp_ip_bytes" ' conn.log

- this gives us all the bytes. But we just need over 40. So we use awk command

jq '."resp_ip_bytes" ' conn.log | awk '$1 > 40' --- this gives us bytes over 40

Here $1 = we are only looking for column 1,

To count the output:

garviel@terra:~$ jq '."resp_ip_bytes" ' conn.log | awk '$1 > 40' | wc -l

177

https://www.geeksforgeeks.org/awk-command-unixlinux-examples/

**20. Which cron log severity code is saved only to the local machine?**

**Flag format: #**

**(Continue to reference your *50-cctc.conf* file from Syslog1)**

21. **The emergency messages (only) on the system are sent to what IP Address?**

    **(Continue to reference your *50-cctc.conf* file from Syslog1)**

22. **Use the log file attached to this for all Whut questions.**

    **How many unique users logged into this machine?**

23. **What is the total amount of time users were logged into the machine?**

    **Round minutes up to the closest number divisible by 10.**

    **Flag format: #*h*,#*m* (Replace the # with a number)**

24. **Identify the Cyber Attack Technique that Balrog is trying on the machine.**

    **HINT: https://attack.mitre.org/**

25. **File: /home/garviel/output.xml**

    **Select every IP address with open (in use) ports using XPATH queries and XPATH axes.**

    **Pipe the result to md5sum for the flag**

    **Sample Output (without piping to MD5SUM)**

26. **Analyze the file to determine when a shell was spawned as a different user and how long it was maintained for.**

    **Provide the :**

    a. **duration the shell was maintained**
    b. **the command used to create it**

c. **number of times they [successfully] escalated**

**Flag Format: #h,#m,command,number of times**

**Round minutes up to the closest number divisible by 10.**

**********

What run levels start the daemon that allows remote connections over port 22?

Flag format: #,#,#,#

NOTE: Use the machine identified in SysV 1 for this question.

cat /etc/inittabcd /etc/init.dCat ssh #look at Default startup for '2 3 4 5'

Default-service

#ORcd etcls -l rc0.d rc1.d rc2.d rc3.d rc4.d rc5.d rc6.d

Flag: 2,3,4,5 (all outputs that have ALL S's to start the last column)

27. Identify the file symbolically-linked to init on the SystemD init machine.

Flag format: /absolute/path

Reminder: Use your Terra machine for these SystemD challenges!

cd /sbin/

Ls -lisa init

/lib/systemd/systemd

```
garviel@terra:/sbin$ ls -lisa init
12407 0 lrwxrwxrwx 1 root root 20 Jul 21  2021 init -> /lib/systemd/systemd
garviel@terra:/sbin$
```

28. What is the default target on the SystemD machine and where is it actually located?

Flag format: name.target,/absolute/path

NOTE: Use the SystemD Machine for this question.

Cd out to main folder and run : find ./ -name default.target

ls -lisa

./lib/systemd/system/default.target

Graphical.target,/lib/systemd/system/graphical.target

29.