**Linux for hackers 6 : Process Management**

What is a process?

A process is simply a program running and using resources ex: terminal, web server, db’s etc..

100s/1000s of processes are running simultaneously. As a hacker/good linux admin should know how to manage their processes and optimize their system. Ex: find and stop antivirus/firewall on target machine.

1)View Processes: **ps**

**Linux kernel**, inner core of the OS that controls nearly everything, assigns a unique process ID to each process simultaneously, as the processes are created. **PID** **is more important than name of the process.**

**ps aux** – shows all processes running on the system for all users.

2)Filtering by process name:

**$msfconsole &**

**$ps aux | grep msfconsole**

The second command will display all the processes that match PostgreSQL, msfconsole program itself /usr/bin/msfconsole

3rd column shows how much CPU it is using and 4th column shows % of system memory being used.

3) Finding the greediest processes with **top**

**$top**

This command will give the list of processes consuming most resources dynamically by default every 10 seconds. When you have top running pressing **H or ?** will bring up a list of interactive commands, pressing **Q will quit top.**

4) Managing processes:

Sometimes we may need a port scanner and vulnerability scanner running, exploit simultaneously. So as a hacker we need to manage the process. Using this command you can setup the priority for a specific process. The values for nice range from -20 to +19, with zero being default value.

Volarant - 1

Movie or video editing

Song -2

Timeline

Description automatically generated with medium confidence

4.1 setting the priority when starting a process:

**$nice -n -10 /bin/importantprocess**

This above command will make the important process a priority and allocate more resources to it.

Let’s say we do NOT want to make it priority, and we can set the lower priority for the new process with the help of the below command. This will allow us to be nice by giving other processes a priority.

**$nice -n -10 /bin/importantprocess**

4.2Changing the priority of a running process with **renice**

This command takes absolute values between -20 and 19 and sets the priority to that particular level, rather than increasing or decreasing from level at which it started.

**$renice -20 6996**

Here 6996 is the PID, and 20 is the value as we discussed above.

We can use **top** utility to change the **nice** value. With the top utility running simply press R key and then supply the PID and the nice value.

5)killing processes:

Kill command has 64 signals, each does something slightly different.

Syntax: **kill -signal PID**

If we don’t provide signal flag it defaults to ‘SIGTERM’ (15)

$**kill -1 6996** – this command is hangup signal restart with same PID

$**kill -9 6996** – this command is a absolute kill signal.

To kill a hypothetical zombie process $**killall -9 zombieprocess**

* Using **top** command we can kill a process by pressing **k** and enter **PID**

6) Running process in background:

**$leafpad newscript &**

Here & will create a process in the background and allow us to run commands on the same terminal to save screen space and resources.

6)Moving a process to the foreground – process from background to foreground

**$fg 1234**

7)scheduling process: backups needs to be scheduled for example by admins or for recon we might want to run a script, find open ports or vulnerabilities in two ways 1)at & 2)crond – scheduling tasks to occur everyday/week/month.

**$at 7:20am**

**at >/root/myscanningscript**

this code snippet will schedule myscanningscript to execute today at 7:20 AM.

**Exercise:**

1. Run the ps command with the aux options on your system and note which process is firstand which is last.

2. Run the top command and note the two processes using the greatest amount of yourresources.

3. Use the kill command to kill the process that uses the most resources.

4. Use the renice command to reduce the priority of a running process to +19.

5. Create a script called myscanning (the content is not important) with a text editor andthen schedule it to run next Wednesday at 1 AM