

SEGURIDAD EN SISTEMAS OPERATIVOS

4º Grado en Informática – Complementos de Ing. del Software
Curso 2017-18

Práctica [1]. Administración de la seguridad en Linux

Sesión [2]. Herramientas básicas de seguridad

Autor¹: Rubén Calvo Villazán

Ejercicio 1.

1º

Actividad de la red con lsof -i:

```
COMMAND PID USER  FD  TYPE DEVICE SIZE/OFF NODE NAME
dhclient 772 root   6u  IPv4 19677    0t0  UDP *:bootpc
megasync 995 root   25u  IPv4 20767    0t0  TCP localhost:6342 (LISTEN)
megasync 995 root   31u  IPv4 45858    0t0  TCP cvi042030.ugr.es:60298->lu5.api.mega.nz:https (ESTABLISHED)
megasync 995 root   34u  IPv4 47830    0t0  TCP cvi042030.ugr.es:57024->lu7.api.mega.nz:http (ESTABLISHED)
```

2º

Para ver si hay un servicio ssh (se está escuchando en el puerto 22):

```
$> lsof -i :22
```

Si hay tráfico saliente ssh:

Si hay una conexión establecida a nuestra máquina por medio de ssh, basta con mirar el fichero: /var/log/auth.log y ver los registros correspondientes a ssh.

3º

```
$> lsof -c init -uroot
```

Usando el comando lsof nos muestra los archivos abiertos por algun proceso, con -c init especificamos que queremos los del proceso init y con -uroot especificamos que queremos todos los correspondientes al usuario root

1 Como autor declaro que los contenidos del presente documento son originales y elaborados por mi. De no cumplir con este compromiso, soy consciente de que, de acuerdo con la “Normativa de evaluación y de calificaciones de los estudiantes de la Universidad de Granada” esto “conllevará la calificación numérica de cero ... independientemente del resto de calificaciones que el estudiante hubiera obtenido ...”

Ejercicio 2.

Ejecutando ps con argumentos distintos cada vez. Hay diferencias a nivel de PID.

```
Terminal - root@kali: ~
File Edit View Terminal Tabs Help

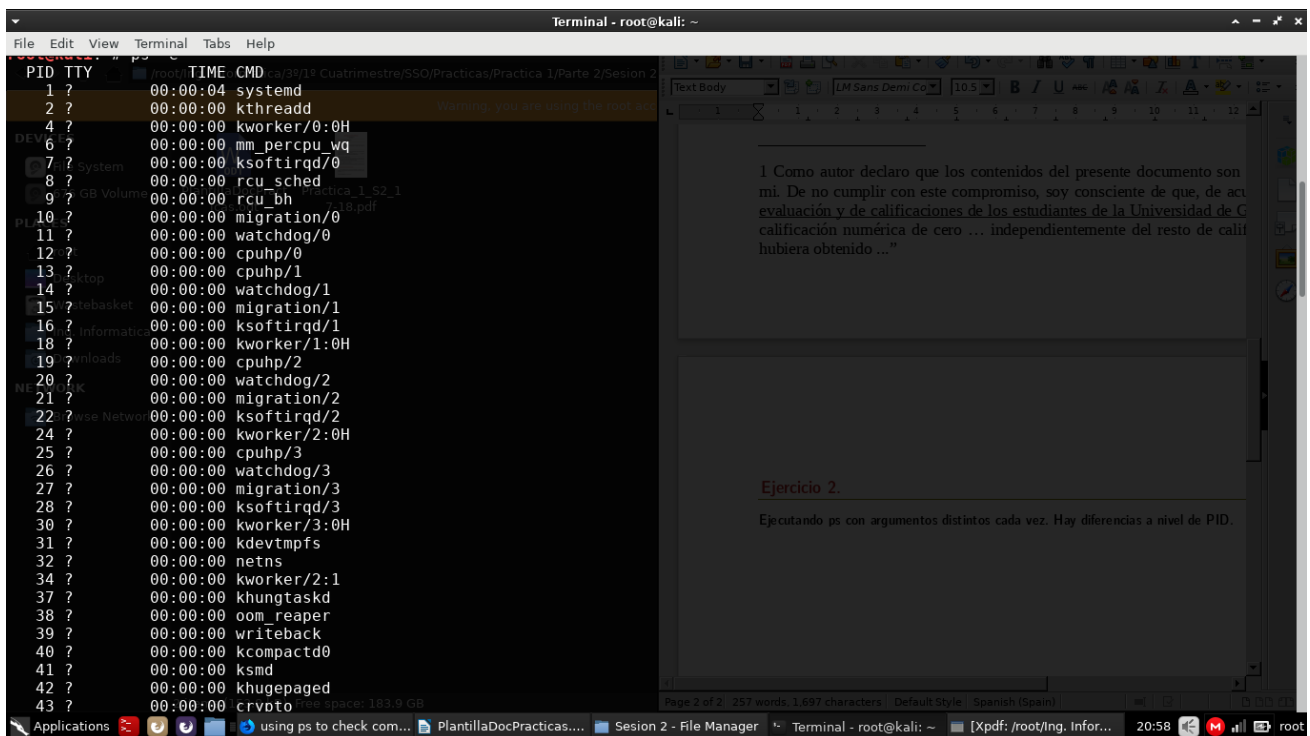
root@kali:~# ps -aux
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1  0.0  0.0 219008 7692 ?        Ss   20:18   0:04 /sbin/init
root         2  0.0  0.0      0      0 ?        Ss   20:18   0:00 [kthreadd]
root         4  0.0  0.0      0      0 ?        S<   20:18   0:00 [kworker/0:0H]
root         6  0.0  0.0      0      0 ?        S<   20:18   0:00 [mm_percpu_wq]
root         7  0.0  0.0      0      0 ?        S<   20:18   0:00 [ksoftirqd/0]
root         8  0.0  0.0      0      0 ?        S<   20:18   0:00 [rcu_sched]
root         9  0.0  0.0      0      0 ?        S<   20:18   0:00 [rcu_bh]
root        10  0.0  0.0      0      0 ?        S<   20:18   0:00 [migration/0]
root        11  0.0  0.0      0      0 ?        S<   20:18   0:00 [watchdog/0]
root        12  0.0  0.0      0      0 ?        S<   20:18   0:00 [cpuhp/0]
root        13  0.0  0.0      0      0 ?        S<   20:18   0:00 [cpuhp/1]
root        14  0.0  0.0      0      0 ?        S<   20:18   0:00 [watchdog/1]
root        15  0.0  0.0      0      0 ?        S<   20:18   0:00 [migration/1]
root        16  0.0  0.0      0      0 ?        S<   20:18   0:00 [ksoftirqd/1]
root        18  0.0  0.0      0      0 ?        S<   20:18   0:00 [kworker/1:0H]
root        19  0.0  0.0      0      0 ?        S<   20:18   0:00 [cpuhp/2]
root        20  0.0  0.0      0      0 ?        S<   20:18   0:00 [watchdog/2]
root        21  0.0  0.0      0      0 ?        S<   20:18   0:00 [migration/2]
root        22  0.0  0.0      0      0 ?        S<   20:18   0:00 [ksoftirqd/2]
root        24  0.0  0.0      0      0 ?        S<   20:18   0:00 [kworker/2:0H]
root        25  0.0  0.0      0      0 ?        S<   20:18   0:00 [cpuhp/3]
root        26  0.0  0.0      0      0 ?        S<   20:18   0:00 [watchdog/3]
root        27  0.0  0.0      0      0 ?        S<   20:18   0:00 [migration/3]
root        28  0.0  0.0      0      0 ?        S<   20:18   0:00 [ksoftirqd/3]
root        30  0.0  0.0      0      0 ?        S<   20:18   0:00 [kworker/3:0H]
root        31  0.0  0.0      0      0 ?        S<   20:18   0:00 [kdevtmpfs]
root        32  0.0  0.0      0      0 ?        S<   20:18   0:00 [netns]
root        34  0.0  0.0      0      0 ?        S<   20:18   0:00 [kworker/2:1]
root        37  0.0  0.0      0      0 ?        S<   20:18   0:00 [khungtaskd]
root        38  0.0  0.0      0      0 ?        S<   20:18   0:00 [oom_reaper]
root        39  0.0  0.0      0      0 ?        S<   20:18   0:00 [writeback]
root        40  0.0  0.0      0      0 ?        S<   20:18   0:00 [kcompactd0]
root        41  0.0  0.0      0      0 ?        SN   20:18   0:00 [ksmd]
root        42  0.0  0.0      0      0 ?        SN   20:18   0:00 [khugepaged]

Ejercicio 2.
Ejecutando ps con argumentos distintos cada vez. Hay diferencias a nivel de PID.
```

```
Terminal - root@kali: ~
File Edit View Terminal Tabs Help

root         46  0.0  0.0      0      0 ?        S<   20:18   0:00 [kblockd]
root         47  0.0  0.0      0      0 ?        S<   20:18   0:00 [devfreq_wq]
root         48  0.0  0.0      0      0 ?        S<   20:18   0:00 [watchdogd]
root         49  0.0  0.0      0      0 ?        S<   20:18   0:00 [kauditd]
root         50  0.0  0.0      0      0 ?        S<   20:18   0:00 [kswapd0]
root         51  0.0  0.0      0      0 ?        S<   20:18   0:00 [bioset]
root         64  0.0  0.0      0      0 ?        S<   20:18   0:00 [kthrotld]
root         66  0.0  0.0      0      0 ?        S<   20:18   0:00 [ipv6_addrconf]
root        101  0.0  0.0      0      0 ?        S<   20:18   0:00 [acpi_thermal_pm]
root        107  0.0  0.0      0      0 ?        S<   20:18   0:00 [ata_sff]
root        163  0.1  0.0      0      0 ?        S<   20:18   0:04 [irq/39-SYN1B81:]
root        164  0.0  0.0      0      0 ?        S<   20:18   0:00 [scsi_ah_0]
root        165  0.0  0.0      0      0 ?        S<   20:18   0:00 [scsi_tmf_0]
root        166  0.0  0.0      0      0 ?        S<   20:18   0:00 [scsi_ah_1]
root        167  0.0  0.0      0      0 ?        S<   20:18   0:00 [scsi_tmf_1]
root        168  0.0  0.0      0      0 ?        S<   20:18   0:00 [scsi_ah_2]
root        169  0.0  0.0      0      0 ?        S<   20:18   0:00 [scsi_tmf_2]
root        170  0.0  0.0      0      0 ?        S<   20:18   0:00 [scsi_ah_3]
root        171  0.0  0.0      0      0 ?        S<   20:18   0:00 [scsi_tmf_3]
root        177  0.0  0.0      0      0 ?        S<   20:18   0:00 [bioset]
root        179  0.0  0.0      0      0 ?        S<   20:18   0:00 [kworker/2:1H]
root        180  0.0  0.0      0      0 ?        S<   20:18   0:00 [kworker/1:1H]
root        181  0.0  0.0      0      0 ?        S<   20:18   0:00 [kworker/0:1H]
root        182  0.0  0.0      0      0 ?        S<   20:18   0:00 [kworker/3:1H]
root        225  0.0  0.0      0      0 ?        S<   20:19   0:00 [jbd2/sda5-8]
root        226  0.0  0.0      0      0 ?        S<   20:19   0:00 [ext4-rsv-conver]
root        269  0.0  0.0 64676 6408 ?        Ss   20:19   0:00 /lib/systemd/systemd-journald
root        287  0.0  0.0 46024 5424 ?        Ss   20:19   0:00 /lib/systemd/systemd-udev
root        351  0.0  0.0      0      0 ?        S<   20:19   0:00 [irq/49-mei_me]
root        354  0.0  0.0      0      0 ?        S<   20:19   0:00 [i915/signal:0]
root        355  0.0  0.0      0      0 ?        S<   20:19   0:00 [i915/signal:1]
root        356  0.0  0.0      0      0 ?        S<   20:19   0:00 [i915/signal:2]
root        357  0.0  0.0      0      0 ?        S<   20:19   0:00 [i915/signal:4]
root        368  0.0  0.0      0      0 ?        S<   20:19   0:00 [cfg80211]
root        373  0.0  0.0      0      0 ?        S<   20:19   0:00 [ath10k_wq]
root        374  0.0  0.0 3.8 kB 0 73.9 GB S<   20:19   0:00 [ath10k_aux_wq]

Ejercicio 2.
Ejecutando ps con argumentos distintos cada vez. Hay diferencias a nivel de PID.
```



Ejercicio 3.

1º

Las vulnerabilidades encontradas son:

- ! No password set for single mode [AUTH-9308]
- ! Can't find any security repository in /etc/apt/sources.list or sources.list.d directory [PKGS-7388]
- ! iptables module(s) loaded, but no rules active [FIRE-4512]

El grado de severidad es medio, algunas sugerencias para solucionarlas son:

Asignar una contraseña al GRUB bootloader y a singlemode.

Separar en particiones /tmp y /var.

Establecer reglas para iptables.

Asignar repositorios seguros en /etc/apt/sources.list para apt.

2º

La vulnerabilidad shellshock tiene de código SHLL-6290, tras ejecutar un analisis con Lynis hacemos:

```
cat /var/log/lynis.log | grep SHLL-6290
```

Si no aparece registro, no tenemos la vulnerabilidad. Para ver otras vulnerabilidades relacionadas con la shell:

```
cat /var/log/lynis.log | grep SHLL
```

```
2017-10-20 21:25:55 Skipped test SHLL-6202 (Check console TTYs)
```

```
2017-10-20 21:25:55 Performing test ID SHLL-6211 (Checking available and valid shells)
```

```
2017-10-20 21:25:55 Performing test ID SHLL-6220 (Checking available and valid shells)
```

```
2017-10-20 21:25:55 Performing test ID SHLL-6230 (Perform umask check for shell configurations)
```

3º

Si tenemos un antivirus instalado, Lynis lo detectará. Podemos ver el resultado del análisis relativo al antivirus con:

```
cat /var/log/lynis.log | grep anti
```

```
2017-10-20 21:51:58 Performing test ID MALW-3280 (Check if anti-virus tool is installed)
```

```
2017-10-20 21:51:58 Test: checking process TmccMac to test for Trend Micro anti-virus (macOS)
```

```
2017-10-20 21:51:58 Result: no commercial anti-virus tools found
```

Ejercicio 4.

1º

Tras realizar un análisis con rkhunter: `rkhunter -check`

Aparece todo **correcto** salvo en las siguientes advertencias.

```
/usr/bin/lwp-request
```

Se soluciona creando o editando `/etc/rkhunter.conf.local`
y añadiendo `PKGMR=DPKG`

Checking for suspicious shared memory segments

Se soluciona añadiendo a whitelist los procesos que usen memoria compartida
`ALLOWIPCPROC=path/to/service`

Checking if SSH root access is allowed

Se soluciona editando el fichero rkhunter:

```
cat /etc/rkhunter.conf | grep ALLOW_SSH_ROOT_USER
```

Cambiando:

```
ALLOW_SSH_ROOT_USER=no
```

Y cambiando en ssh:

```
cat /etc/ssh/sshd_config | grep PermitRootLogin
```

Editando:

```
PermitRootLogin no
```