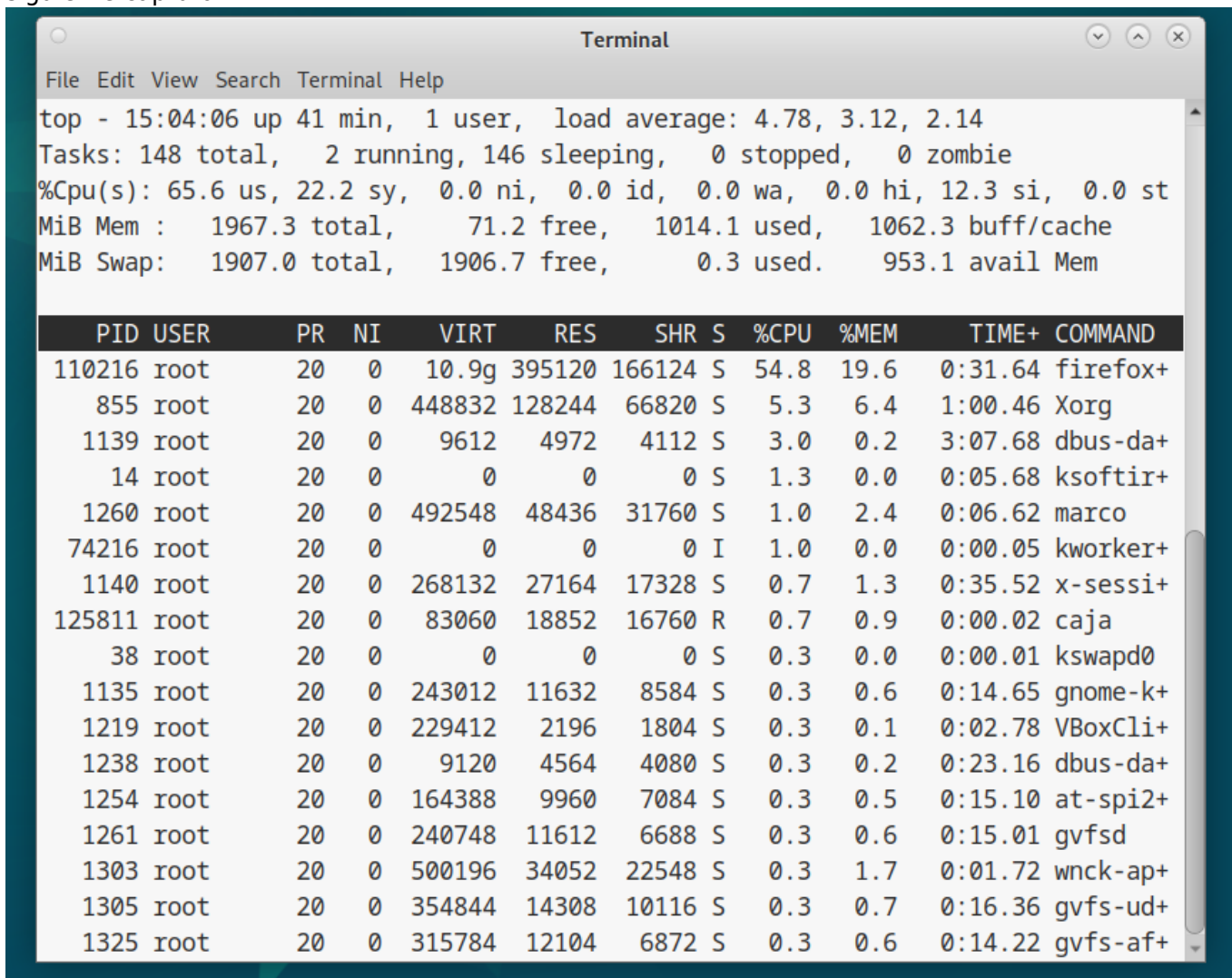


# [Fort]Práctica 3: Securizando Aplicaciones

## 1. Abre un navegador y realiza una descarga

### Revisa con el TOP el consumo de CPU

Firefox (Proceso 110216) está consumiendo en torno al 60% de la CPU como podemos ver en la siguiente captura:



```
top - 15:04:06 up 41 min,  1 user,  load average: 4.78, 3.12, 2.14
Tasks: 148 total,  2 running, 146 sleeping,  0 stopped,  0 zombie
%Cpu(s): 65.6 us, 22.2 sy,  0.0 ni,  0.0 id,  0.0 wa,  0.0 hi, 12.3 si,  0.0 st
MiB Mem :  1967.3 total,   71.2 free,  1014.1 used,  1062.3 buff/cache
MiB Swap:  1907.0 total,  1906.7 free,    0.3 used.  953.1 avail Mem

  PID USER      PR  NI   VIRT   RES   SHR  S  %CPU  %MEM     TIME+ COMMAND
 110216 root      20   0  10.9g 395120 166124 S  54.8  19.6   0:31.64 firefox+
   855 root      20   0 448832 128244  66820 S   5.3   6.4   1:00.46 Xorg
   1139 root      20   0   9612   4972   4112 S   3.0   0.2   3:07.68 dbus-da+
    14 root      20   0     0     0     0 S   1.3   0.0   0:05.68 ksofttir+
   1260 root      20   0 492548 48436  31760 S   1.0   2.4   0:06.62 marco
  74216 root      20   0     0     0     0 I   1.0   0.0   0:00.05 kworker+
   1140 root      20   0 268132 27164  17328 S   0.7   1.3   0:35.52 x-sessi+
125811 root      20   0   83060  18852  16760 R   0.7   0.9   0:00.02 caja
    38 root      20   0     0     0     0 S   0.3   0.0   0:00.01 kswapd0
   1135 root      20   0 243012  11632   8584 S   0.3   0.6   0:14.65 gnome-k+
   1219 root      20   0 229412   2196   1804 S   0.3   0.1   0:02.78 VBoxCli+
   1238 root      20   0   9120   4564   4080 S   0.3   0.2   0:23.16 dbus-da+
   1254 root      20   0 164388   9960   7084 S   0.3   0.5   0:15.10 at-spi2+
   1261 root      20   0 240748  11612   6688 S   0.3   0.6   0:15.01 gvfsd
   1303 root      20   0 500196  34052  22548 S   0.3   1.7   0:01.72 wnck-ap+
   1305 root      20   0 354844  14308  10116 S   0.3   0.7   0:16.36 gvfs-ud+
   1325 root      20   0 315784  12104   6872 S   0.3   0.6   0:14.22 gvfs-af+
```

### Usa cputlimit para reducir el consumo de CPU a 1/5 del que usa

Para realizar esto ejecutaremos los siguientes comandos:

```
cd /sys/fs/cgroup/
mkdir firefox #Creamos el cgroup para firefox
cd firefox
```

```
root@fso2025:~/Downloads# cd /sys/fs/cgroup/  
root@fso2025:/sys/fs/cgroup# mkdir firefox  
root@fso2025:/sys/fs/cgroup# cd firefox/  
root@fso2025:/sys/fs/cgroup/firefox# ls  
cgroup.controllers      cpuset.cpus.effective  memory.min  
cgroup.events          cpuset.cpus.partition  memory.numa_stat  
cgroup.freeze          cpuset.mems            memory.oom.group  
cgroup.kill            cpuset.mems.effective  memory.peak  
cgroup.max.depth       cpu.stat               memory.pressure  
cgroup.max.descendants  cpu.weight             memory.reclaim  
cgroup.pressure        cpu.weight.nice        memory.stat  
cgroup.procs           io.max                 memory.swap.current  
cgroup.stat            io.pressure            memory.swap.events  
cgroup.subtree_control io.stat                memory.swap.high  
cgroup.threads         io.weight              memory.swap.max  
cgroup.type            memory.current         memory.zswap.current  
cpu.idle               memory.events          memory.zswap.max  
cpu.max                memory.events.local    pids.current  
cpu.max.burst          memory.high             pids.events  
cpu.pressure           memory.low              pids.max  
cpuset.cpus            memory.max              pids.peak  
root@fso2025:/sys/fs/cgroup/firefox# █
```

Tras eso añadimos firefox al cgroup y procedemos a limitar su consumo de CPU

```
echo 110216 > cgroup.procs#Metemos el proceso de firefox en el cgroup  
echo 200000 1000000 > cpu.max #Limitamos el uso de CPU al 20%
```

```
root@fso2025:/sys/fs/cgroup/firefox# echo 110216 > cgroup.procs  
root@fso2025:/sys/fs/cgroup/firefox# echo 200000 1000000 > cpu.max  
root@fso2025:/sys/fs/cgroup/firefox# █
```

## Resultado

Como se puede observar, el consumo ha bajado y ahora consume un 20% máximo.

```
top - 15:17:43 up 54 min,  1 user,  load average: 0.08, 0.51, 1.27
Tasks: 145 total,  1 running, 144 sleeping,  0 stopped,  0 zombie
%Cpu(s): 12.8 us,  4.7 sy,  0.0 ni, 79.1 id,  0.0 wa,  0.0 hi,  3.4 si,  0.0 st
MiB Mem :  1967.3 total,   700.8 free,  1033.6 used,   439.1 buff/cache
MiB Swap:  1907.0 total,  1865.5 free,   41.5 used.   933.7 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
110216	root	20	0	3023872	337904	112236	S	20.3	16.8	3:16.39	firefox+
855	root	20	0	490648	109572	51580	S	0.7	5.4	1:15.46	Xorg
14	root	20	0	0	0	0	S	0.3	0.0	0:08.80	ksoftir+
15	root	20	0	0	0	0	I	0.3	0.0	0:05.32	rcu_pre+
1209	root	20	0	228896	1548	1276	S	0.3	0.1	0:01.28	VBoxCli+
1219	root	20	0	229412	1484	1212	S	0.3	0.1	0:04.24	VBoxCli+
1	root	20	0	103904	11088	6424	S	0.0	0.6	0:04.73	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kthreadd
3	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	rcu_gp
4	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	rcu_par+
5	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	slub_fl+
6	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	netns
7	root	20	0	0	0	0	I	0.0	0.0	0:01.66	kworker+
10	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	mm_perc+
11	root	20	0	0	0	0	I	0.0	0.0	0:00.00	rcu_tas+
12	root	20	0	0	0	0	I	0.0	0.0	0:00.00	rcu_tas+
13	root	20	0	0	0	0	I	0.0	0.0	0:00.00	rcu_tas+

## 2. Crea un container Debian y arráncalo

Para arrancar un container debian usamos el siguiente comando:

```
lxc-create -t debian -n deb
```

### Establece una contraseña para el usuario root del container

Para ello debemos entrar primero a la máquina con el siguiente comando para abrir la terminal del contenedor:

```
lxc-start #Arrancamos la máquina
lxc-attach -n deb /bin/sh #Llamamos a la terminal
```

```
root@fso2025:~# lxc-st
lxc-start lxc-stop
root@fso2025:~# lxc-start -n deb
root@fso2025:~# lxc-attach -n deb /bin/sh
# █
```

Tras eso procedemos a establecer la contraseña del root con el siguiente comando:

```
passwd root
```

```
root@fso2025:~# lxc-start -n deb
root@fso2025:~# lxc-attach -n deb /bin/sh
# passwd root
New password:
Retype new password:
passwd: password updated successfully
#
```

### Añade 3 usuarios

Desde el mismo sitio que hemos establecido la contraseña para el root procedemos a crear los 3 usuarios:

```
su - root
useradd -m -s /bin/bash usuario1
useradd -m -s /bin/bash usuario2
useradd -m -s /bin/bash usuario3
```

```
# su - root
root@deb:~# sudo useradd -m -s /bin/bash usuario1
-bash: sudo: command not found
root@deb:~# useradd -m -s /bin/bash usuario1
root@deb:~# useradd -m -s /bin/bash usuario2
root@deb:~# useradd -m -s /bin/bash usuario3
root@deb:~# █
```

### Instala apache 2 y ssh en el container

Comenzamos instalando apache 2:

```
root@deb:~# apt install apache2
Reading package lists... Done
Building dependency tree... Done
The following additional packages will be installed:
 apache2-bin apache2-data apache2-utils libapr1 libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap libbrotli1
 libcurl4 libexpat1 libgdbm-compat4 libgdbm6 libicu72 libjansson4 libldap-2.5-0 libldap-common liblua5.3-0
 libnghttp2-14 libperl5.36 libpsl5 librtmp1 libsasl2-2 libsasl2-modules libsasl2-modules-db libsasl2-modules-gssapi-mit
 libsasl2-modules-gssapi-heimdal libsasl2-modules-ldap libsasl2-modules-otp libsasl2-modules-sql perl-doc
 libterm-readline-gnu-perl | libterm-readline-perl-perl make libtap-harness-archive-perl
Suggested packages:
 apache2-doc apache2-suexec-pristine | apache2-suexec-custom www-browser gdbm-l10n libsasl2-modules-gssapi-mit
 | libsasl2-modules-gssapi-heimdal libsasl2-modules-ldap libsasl2-modules-otp libsasl2-modules-sql perl-doc
 libterm-readline-gnu-perl | libterm-readline-perl-perl make libtap-harness-archive-perl
The following NEW packages will be installed:
 apache2 apache2-bin apache2-data apache2-utils libapr1 libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap
 libbrotli1 libcurl4 libexpat1 libgdbm-compat4 libgdbm6 libicu72 libjansson4 libldap-2.5-0 libldap-common
 liblua5.3-0 libnghttp2-14 libperl5.36 libpsl5 librtmp1 libsasl2-2 libsasl2-modules libsasl2-modules-db
 libsasl2-modules-gssapi-mit libsasl2-modules-gssapi-heimdal libsasl2-modules-ldap libsasl2-modules-otp
 libsasl2-modules-sql perl-doc perl libterm-readline-gnu-perl | libterm-readline-perl-perl make libtap-harness-archive-perl
0 upgraded, 33 newly installed, 0 to remove and 0 not upgraded.
Need to get 22.3 MB of archives.
```

Tras eso procedemos a instalar SSH:

```
root@deb:~# apt install ssh
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
 ssh
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 174 kB of archives.
After this operation, 187 kB of additional disk space will be used.
Get:1 http://deb.debian.org/debian stable/main amd64 ssh all 1:9.2p1-2+deb12u4 [174 kB]
Fetched 174 kB in 0s (946 kB/s)
debconf: delaying package configuration, since apt-utils is not installed
Selecting previously unselected package ssh.
(Reading database ... 12070 files and directories currently installed.)
Preparing to unpack ../ssh_1%3a9.2p1-2+deb12u4_all.deb ...
Unpacking ssh (1:9.2p1-2+deb12u4) ...
Setting up ssh (1:9.2p1-2+deb12u4) ...
root@deb:~#
```

## Cambia la dirección de red en el container para usar una dirección de red estática

Para establecer una IP estática usamos el siguiente comando:

```
sudo ifconfig eth0 10.0.3.68 netmask 255.255.255.0 up
```

```
root@deb:~# ifconfig eth0 10.0.3.68 netmask 255.255.255.0 up
root@deb:~# ping google.es
PING google.es (142.250.200.131) 56(84) bytes of data.
64 bytes from mad41s14-in-f3.1e100.net (142.250.200.131): icmp_seq=1 ttl=114 time=26.4 ms
64 bytes from mad41s14-in-f3.1e100.net (142.250.200.131): icmp_seq=2 ttl=114 time=26.8 ms
^C
--- google.es ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 26.444/26.641/26.839/0.197 ms
root@deb:~# █
```

## Cambial el puerto de SSH al 222

Vamos a la ruta /etc/ssh y modificamos el archivo sshd\_config:

```
GNU nano 7.2 sshd_config *
# This is the sshd server system-wide configuration file.  See
# sshd_config(5) for more information.

# This sshd was compiled with PATH=/usr/local/bin:/usr/bin:/bin:/usr/games

# The strategy used for options in the default sshd_config shipped with
# OpenSSH is to specify options with their default value where
# possible, but leave them commented.  Uncommented options override the
# default value.

Include /etc/ssh/sshd_config.d/*.conf

Port 222
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key

^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location   M-U Undo
^X Exit      ^R Read File  ^\ Replace    ^U Paste      ^J Justify    ^/ Go To Line  M-E Redo
```

## Modifica el archivo de configuración del container

```
GNU nano 7.2 /var/lib/lxc/deb/config *
# (Be aware this has security implications)

lxc.net.0.type = veth
lxc.net.0.hwaddr = 00:16:3e:3b:74:2a
lxc.net.0.link = lxcbr0
lxc.net.0.flags = up
lxc.apparmor.profile = generated
lxc.apparmor.allow_nesting = 1
lxc.rootfs.path = dir:/var/lib/lxc/deb/rootfs

# Common configuration
lxc.include = /usr/share/lxc/config/debian.common.conf

# Container specific configuration
lxc.tty.max = 4
lxc.uts.name = deb
lxc.arch = amd64
lxc.pty.max = 1024
lxc.start.auto = 1
lxc.start.delay = 3
```

## Crea un csgroup

From:  
<https://knoppia.net/> - Knoppia

Permanent link:  
[https://knoppia.net/doku.php?id=master\\_cs:fortificacion:p3&rev=1739890276](https://knoppia.net/doku.php?id=master_cs:fortificacion:p3&rev=1739890276)

Last update: **2025/02/18 14:51**

