



239-[LX]-Lab - Procesos administrativos

Datos Generales:

Nombre: Tomás Alfredo Villaseca Constantinescu

País: Chile

Fecha: 08/09/2023

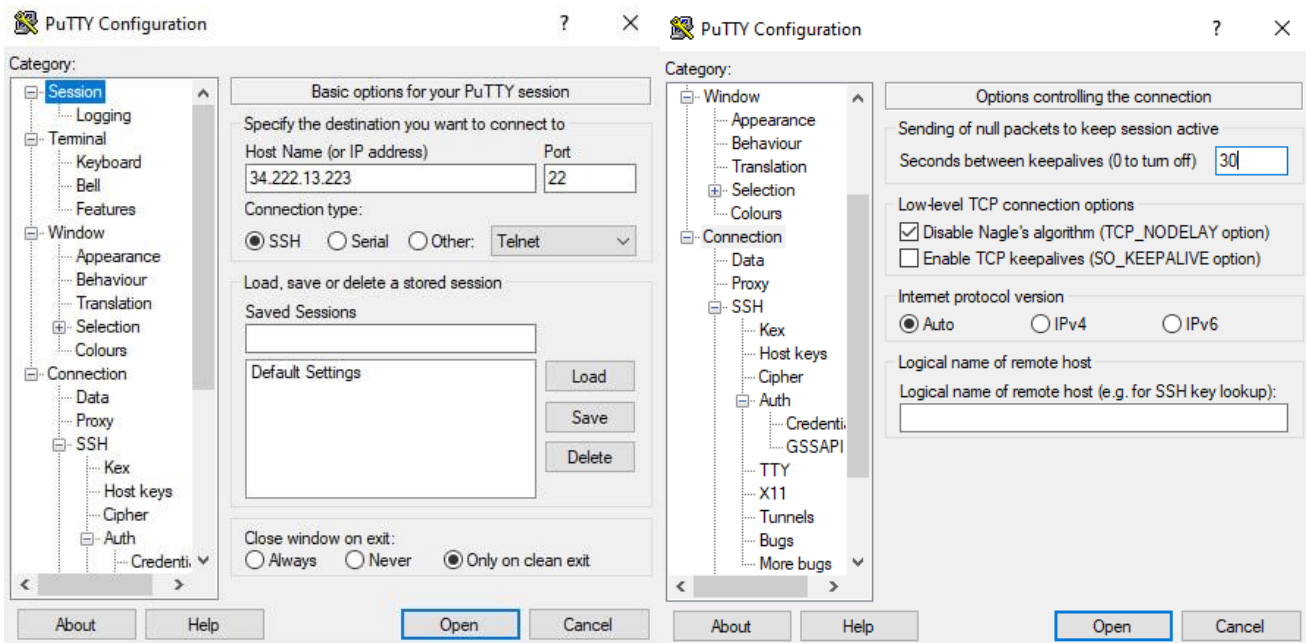
Contacto: tomas.villaseca.c@gmail.com

En este laboratorio, hará lo siguiente:

- Crear un archivo de registro nuevo para las listas de procesos
- Utilizar el comando **top**
- Establecer una tarea repetitiva que ejecute los comandos de auditoría anteriores una vez al día

Tarea 1: conectarse a una instancia EC2 de Amazon Linux mediante SSH

1. Abrir Putty.exe: Se ingresa dirección IPv4 de la instancia EC2 en la sección Session.
2. En la sección Connection → SSH → Auth → Credentials se ingresa el archivo PPK descargado anteriormente.
3. En la sección Connection se establece **Seconds between keepalive en 30 (el valor predeterminado es 0).**



4. Se hace click en "Open" para validar y conectarse al Host.

```
ec2-user@ip-10-0-10-186:~  
login as: ec2-user  
Authenticating with public key "imported-openssh-key"  
  
  _ | _ | _ )  
  _ | ( _ /   Amazon Linux 2 AMI  
  _ | \ _ | _ |  
  
https://aws.amazon.com/amazon-linux-2/  
[ec2-user@ip-10-0-10-186 ~]$
```

Tarea 2: Crear una lista de procesos

sudo = permite ejecutar comandos con privilegios de superusuario.

ps = muestra una lista de los procesos que se están ejecutando en el sistema.

- -a = muestra todos los procesos
- -u = muestra información sobre los procesos del usuario especificado
- -x = muestra procesos que no están asociados a una terminal

grep = busca patrones de texto en archivos

- -v = imprime todas las líneas que no coinciden con el patrón

tee = redirige la salida de un comando al standard output y a uno o más archivos

- Se utiliza para guardar una copia de la salida de un comando

```
ec2-user@ip-10-0-10-213:~/companyA
[ec2-user@ip-10-0-10-213 companyA]$ pwd
/home/ec2-user/companyA
[ec2-user@ip-10-0-10-213 companyA]$ sudo ps -aux | grep -v root | sudo tee SharedFolders/processes.csv
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
dbus      1698  0.0  0.4  58248 4120 ?        Ss   21:03   0:00 /usr/bin/dbus-daemon --system --address=systemd: --nofork --nopidfile --systemd-activation
rpc       1699  0.0  0.3  67256 3100 ?        Ss   21:03   0:00 /sbin/rpcbind -w
libstor+ 1703  0.0  0.2  12628 1940 ?        Ss   21:03   0:00 /usr/bin/lsm -d
rngd      1731  0.0  0.4  94136 4492 ?        Ss   21:03   0:00 /sbin/rngd -f --fill-watermark=0 --exclude=jitter
chrony    1733  0.0  0.3  120344 3172 ?        S   21:03   0:00 /usr/sbin/chronyd -F 2
postfix   2136  0.0  0.6  90388 6736 ?        S   21:03   0:00 pickup -l -t unix -u
postfix   2137  0.0  0.6  90464 6756 ?        S   21:03   0:00 qmgr -l -t unix -u
ec2-user  5559  0.0  0.4  150632 4636 ?        S   21:03   0:00 sshd: ec2-user@pts/0
ec2-user  5665  0.0  0.4  124736 3972 pts/0    Ss   21:03   0:00 -bash
[ec2-user@ip-10-0-10-213 companyA]$ ls
absolute_mode_file  CEO  Documents  Employees  HR  Management  Roster.csv  Sales  SharedFolders  Shipping  symbolic_mode_file
[ec2-user@ip-10-0-10-213 companyA]$ ls SharedFolders
logins.csv  processes.csv
[ec2-user@ip-10-0-10-213 companyA]$
```

cat = mostrar el contenido de un archivo

```
ec2-user@ip-10-0-10-213:~/companyA/SharedFolders
[ec2-user@ip-10-0-10-213 companyA]$ cd SharedFolders
[ec2-user@ip-10-0-10-213 SharedFolders]$ ls
logins.csv  processes.csv
[ec2-user@ip-10-0-10-213 SharedFolders]$ cat processes.csv
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
dbus      1698  0.0  0.4  58248 4120 ?        Ss   21:03   0:00 /usr/bin/dbus-daemon --system --address=systemd: --nofork --nopidfile --systemd-activation
rpc       1699  0.0  0.3  67256 3100 ?        Ss   21:03   0:00 /sbin/rpcbind -w
libstor+ 1703  0.0  0.2  12628 1940 ?        Ss   21:03   0:00 /usr/bin/lsm -d
rngd      1731  0.0  0.4  94136 4492 ?        Ss   21:03   0:00 /sbin/rngd -f --fill-watermark=0 --exclude=jitter
chrony    1733  0.0  0.3  120344 3172 ?        S   21:03   0:00 /usr/sbin/chronyd -F 2
postfix   2136  0.0  0.6  90388 6736 ?        S   21:03   0:00 pickup -l -t unix -u
postfix   2137  0.0  0.6  90464 6756 ?        S   21:03   0:00 qmgr -l -t unix -u
ec2-user  5559  0.0  0.4  150632 4636 ?        S   21:03   0:00 sshd: ec2-user@pts/0
ec2-user  5665  0.0  0.4  124736 3972 pts/0    Ss   21:03   0:00 -bash
[ec2-user@ip-10-0-10-213 SharedFolders]$
```


Tarea 3: Enumerar los procesos mediante el comando "top"

top = muestra una lista de los procesos que se están ejecutando en el sistema en tiempo real

ec2-user@ip-10-0-10-213:~/companyA/SharedFolders

```
top - 21:22:58 up 19 min,  1 user,  load average: 0.00, 0.00, 0.00
Tasks:  87 total,   1 running,  47 sleeping,   0 stopped,   0 zombie
%Cpu(s):  0.2 us,  0.0 sy,  0.0 ni, 99.8 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
KiB Mem :  966816 total,  279324 free,   86792 used,  600700 buff/cache
KiB Swap:   0 total,    0 free,    0 used.  731936 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
6051	ec2-user	20	0	168964	4536	3844	R	0.3	0.5	0:00.01	top
1	root	20	0	123740	5592	3864	S	0.0	0.6	0:01.60	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kthreadd
4	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/0:0H
6	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	mm_percpu_wq
7	root	20	0	0	0	0	S	0.0	0.0	0:00.03	ksoftirqd/0
8	root	20	0	0	0	0	I	0.0	0.0	0:00.05	rcu_sched
9	root	20	0	0	0	0	I	0.0	0.0	0:00.00	rcu_bh
10	root	rt	0	0	0	0	S	0.0	0.0	0:00.00	migration/0
11	root	rt	0	0	0	0	S	0.0	0.0	0:00.00	watchdog/0
12	root	20	0	0	0	0	S	0.0	0.0	0:00.00	cpuhp/0
13	root	20	0	0	0	0	S	0.0	0.0	0:00.01	cpuhp/1
14	root	rt	0	0	0	0	S	0.0	0.0	0:00.00	watchdog/1
15	root	rt	0	0	0	0	S	0.0	0.0	0:00.20	migration/1
16	root	20	0	0	0	0	S	0.0	0.0	0:00.02	ksoftirqd/1
18	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/1:0H
20	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kdevtmpfs
21	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	netns
31	root	20	0	0	0	0	I	0.0	0.0	0:00.18	kworker/u4:2
33	root	20	0	0	0	0	I	0.0	0.0	0:00.10	kworker/1:2
121	root	20	0	0	0	0	S	0.0	0.0	0:00.00	khungtaskd
203	root	20	0	0	0	0	S	0.0	0.0	0:00.00	oom reaper

Tarea 4: Crear un trabajo cron

En esta tarea se creará un **cron job** que generará un archivo de auditoría con ##### para cubrir todos los archivos .csv:

crontab = permite a los usuarios programar tareas para que se ejecuten en un momento específico.

- Funciona creando un archivo llamado crontab en el directorio inicial del usuario
- Archivo crontab contiene una lista de comandos que se ejecutarán en horario específicos.
- -e = abre el archivo de cron del usuario actual para su edición
- -l = muestra el contenido del archivo de cron del usuario actual

```
ec2-user@ip-10-0-10-213:~/companyA
[ec2-user@ip-10-0-10-213 companyA]$ pwd
/home/ec2-user/companyA
[ec2-user@ip-10-0-10-213 companyA]$ sudo crontab -e
no crontab for root - using an empty one
crontab: installing new crontab
[ec2-user@ip-10-0-10-213 companyA]$ ls
absolute_mode_file  CEO  Documents  Employees  HR  Management  Roster.csv  Sales  SharedFolders  Shipping  symbolic_mode_file
[ec2-user@ip-10-0-10-213 companyA]$ sudo crontab -l

SHELL=/bin/bash
PATH=/usr/bin:/bin:/usr/local/bin
MAILTO=root
0 * * * * ls -la $(find .) | sed -e 's/..csv/#####.csv/g' > /home/ec2-user/companyA/SharedFolders/filteredAudit.csv
[ec2-user@ip-10-0-10-213 companyA]$
```

Laboratorio Completado

