

Prácticas con diversas herramientas de administración

Laboratorio de Administración y Gestión de Redes y Sistemas 2019-2020
Grado en Ingeniería Telemática
Escuela Técnica Superior de Ingeniería de Telecomunicación
Universidad Rey Juan Carlos

Observaciones

- Crea el fichero `~/lagrs/practica03.txt`, que contendrá la memoria que escribas sobre este bloque de prácticas. En la primera línea, indica tu nombre y login.

Práctica 3.1. screen

El objetivo de esta práctica es que uses *screen* para mantener demonios corriendo sin terminales abiertos, así como multiplexar sesiones.

Prepararás dos sesiones de screen:

- Una con dos ventanas, una para el script *tictac* y otra para la orden *vmstat*.
- Otra con una única ventana, para la orden *top*.

Hazlo según los siguientes pasos:

1. Empieza familiarizándote con screen, probando los ejemplos de las transparencias. No hace falta que documentes esto en la memoria.
2. Crea el fichero `~/lagrs/practica03/tictactULOGIN` y pega en él este script. (TULOGIN será tu nombre de usuario en el laboratorio).

```
#!/bin/bash
fichero_salida=/tmp/log.$USER.txt
while true
```

```
do
    sleep 1
    echo -n "tic" >> $fichero_salida
    sleep 1
    echo " tac" >> $fichero_salida
done
```

Pruébalo usando `tail -f` (que escribe en la salida estándar las últimas líneas de un fichero, y queda a la espera de que haya cambios en el fichero, para escribirlos también).

3. Busca una máquina linux del laboratorio a la que puedas entrar por ssh.
4. Copia `tictac` al directorio `/tmp/` de la máquina remota.
5. Entra por ssh en la máquina remota.
6. Haz lo necesario para invocar a `tictacTULOGIN` y que quede funcionando en una sesión de screen.
7. Deja `tictacTULOGIN` corriendo y sal de la máquina remota.
8. Vuelve a entrar por ssh y recupera el control de la sesión de `screen` con `tictacTULOGIN`.
9. Crea dentro de la sesión de `screen` otra ventana. Ejecuta en ella `vmstat 2`.
10. Muévete entre una ventana y otra.
11. Renombra las ventanas. La primera se llamará *reloj*. La segunda, *control de memoria*.
12. Deja ambas corriendo. Sal de esta sesión de screen, crea otra sesión de screen y ejecuta en ella la orden `top`.
13. Pon a esta última ventana como nombre *top*.
14. Sal de la máquina remota, dejando todo en marcha (`tictacTULOGIN`, `vmstat` y `top`).
15. Vuelva a entrar y comprueba que sabes recuperar las tres ventanas (las dos ventanas de la primera sesión y la ventana de la segunda sesión).
16. Mata con `Ctrl C` los procesos de las tres ventanas. Cierra todas las ventanas. Comprueba que están cerradas todas las ventanas y todas las sesiones de screen. Cierra la sesión de ssh.

Práctica 3.2. Túnel ssh inverso

En este ejercicio probarás el uso de un túnel inverso de ssh, que te permitirá acceder desde cualquier lugar de internet a un servicio en una dirección privada, detrás de un NAT, sin necesidad de *port forwarding* (*abrir puertos*). Usarás tu host del laboratorio como proxy, que como sabes tiene una dirección IP pública (aunque para este ejercicio bastaría cualquier IP accesible para el cliente, podría ser por ejemplo una IP privada del laboratorio).

Usaremos un servicio de prueba, similar a un *holamundo*. Son un par de scripts en python, cliente y servidor. El servidor recibe un número arábigo y devuelve el número romano correspondiente.

1. Descarga `romanserver.py` y `romanclient.py` en `~/bin`, dale permisos de ejecución y comprueba que el directorio `~/bin` está incluido en tu variable de entorno `PATH`.
2. Prueba `romanserver.py` y `romanclient.py`, en cualquier puerto TCP de tu host del laboratorio. Encontrarás sus instrucciones de uso al lanzarlos sin argumentos.
3. Usando `vagrant`, lanza la máquina virtual `vbox` de la práctica 1.
4. Para poder usar `romanserver.py` en `vbox`, instala python.
5. Copia `romanserver.py` en cualquier directorio de `vbox` y lánzalo en el puerto 8000 de tu máquina virtual.
6. Usando un túnel inverso de ssh, haz que este servicio esté disponible en el puerto 9000 de tu host del laboratorio.
7. Prueba el túnel usando `romanclient.py` en tu host.
8. Lanza `netstat` en ambas máquinas y analiza brevemente el resultado, en lo relativo a este ejercicio.

Práctica 3.3. netstat

Abre una sesión ssh entre `vbox` y tu puesto del laboratorio. Lanza `netstat` en el cliente y analiza el resultado. Lánzalo en el servidor y analiza el resultado

Práctica 3.4. recode

Usando `file` y `recode`, cambia la codificación de un par de ficheros de texto y comprueba el resultado.

Revisión de los nombres

Ejecuta el script `~mortuno/revisa practicas lagrs` para comprobar que los nombres de los ficheros son correctos.