

# Laboratorio de administración y gestión de redes y sistemas

## Examen práctico. 13 de junio de 2023.

Grado en Ingeniería Telemática, Universidad Rey Juan Carlos

---

Preparativos:

- Crea un directorio `~/lagrs.junio.23/`, donde escribirás todo lo indicado a continuación.

### Ejercicio 1 (5 puntos)

El uso más frecuente de la orden `ssh` es abrir una sesión de shell interactiva en una máquina remota. Pero también sirve para ejecutar un único comando en una máquina remota, mostrando el resultado de la ejecución en la máquina local. Por ejemplo, el comando `ssh juan@alpha uptime` ejecutaría la orden `uptime` en la máquina `alpha` con el usuario `juan` y nos mostraría en nuestra máquina el resultado.

Así, para saber el tiempo que llevan encendidas las máquinas `alpha`, `beta` y `gamma`, podríamos ejecutar un script de shell como este:

```
#!/bin/bash
ssh juan@alpha uptime
ssh juan@beta uptime
ssh juan@gamma uptime
```

Teniendo esto en cuenta, escribe en el fichero `~/lagrs.junio.23/tiempos.TULOGIN.py` un script en Python 3 según la siguiente especificación

1. Este script de Python generará en la salida estándar un script de shell similar al anterior, que nos muestre el tiempo que llevan encendidas todas las máquinas *de profesor* de los laboratorios del campus de Fuenlabrada
2. Para saber cuáles son las máquinas *de profesor*, esto es, en qué máquinas se debe ejecutar el comando, tu programa consultará el fichero `/etc/hosts` de tu máquina y seleccionara todas aquellas cuyo nombre comience por `f-` y acabe en `-pc00`
3. Para que el script de shell resultante sea usable, es necesario que se pueda abrir una sesión en una máquina remota sin escribir contraseña. En otras palabras, que tu práctica 1.9 funcione correctamente.
4. El nombre del usuario estará indicado en la variable global `USUARIO`, en una de las primeras líneas de tu programa en Python. Dale como valor tu propio nombre de usuario.
5. El script tendrá en la segunda línea un comentario indicando cuántas máquinas de profesor hay en los laboratorios del campus de Fuenlabrada. Ejemplo:

```
#!/bin/bash
# En el campus hay 12 máquinas de profesor
ssh jperez@f-12102-pc00 uptime
ssh jperez@f-12108-pc00 uptime
ssh jperez@f-13103-pc00 uptime
... (etc)
```

6. Para hacer este script realista, habría que contemplar que alguna de las máquinas puede estar inaccesible. Pero ignora esto, supón que puedes entrar en todas ellas.
7. Cuando acabes el ejercicio, avisa al profesor, muéstrale tu script, ejecútalo, redirige la salida a un fichero de texto y ejecuta el script de shell resultante.

## Ejercicio 2 (5 puntos)

Prepara un contenedor según la siguiente especificación

1. El contenedor estará basado en una imagen llamada `ex23`.
2. El contenedor estará construido a partir de `ubuntu 22.04`, con los paquetes actualizados a la última versión disponible. Tendrá `tmux` instalado. Y ningún otro paquete ni ninguna otra configuración distinta a la de un contenedor `docker` por omisión. Si añades paquetes o configuración adicional, tendrás una penalización en la nota.
3. El nombre del contenedor será el nombre de la imagen, añadiendo como prefijo las primeras 4 letras de tu nombre de usuario en el laboratorio, y añadiendo el sufijo `01`. Ejemplo: Si tu nombre de usuario fuera `jperez`, el contenedor se llamaría `jporex2301`.
4. En el contenedor habrá un usuario con el mismo nombre que tu usuario del laboratorio.
5. El `id` de este usuario será el mismo que tu `id` en el laboratorio.
6. Al ejecutarse el contenedor, tendremos abierta una sesión de `shell` de este usuario.
7. El usuario podrá lanzar `tmux` si lo desea. Pero tendrá preparada una configuración personalizada: queremos que la tecla `bind` no sea `ctrl b` sino `ctrl a`, tal y como vimos en clase y se describe en las transparencias.
8. Mediante un montaje `bind`, dentro del directorio `~/tmp` del contenedor estará montado el directorio `/tmp/` del servidor de contenedores.
9. Los ficheros necesarios para todo esto seguirán el mismo convenio que hemos venido usando durante toda la asignatura.

```
~/lagrs.junio.23/ex23/construye.sh
~/lagrs.junio.23/ex23/lanza_jperexa01.sh
~/lagrs.junio.23/ex23/context/Dockerfile
~/lagrs.junio.23/ex23/context/entrypoint.sh
~/lagrs.junio.23/ex23/context/tmux.conf
```

(Reemplazando `jper` por las primeras 4 letras de tu `login`)

10. Observa que el fichero de configuración de `tmux` es un fichero oculto, empieza por punto, cuando esté dentro del contenedor debe empezar por punto. Pero cuando lo prepares en el directorio `context`, no debe estar oculto, su nombre no debe empezar por punto.
11. Enséñaselo al profesor cuando acabes.
12. El profesor te pedirá que lances `tmux`, que abras varios paneles, los recoloces, te muevas entre ellos, te desasocias de esta sesión de `tmux` y te vuelvas a asociar.