

## Práctica 9. Programación de tareas en Windows y Linux.

Uno de las ventajas de la **programación de tareas** para un **Administrador de Sistemas** es que permite, entre otras cosas, la **automatización de determinados trabajos repetitivos** sin la intervención de este con la comodidad que ello representa.

Como **material de trabajo** apuntes de la UD4 y clones enlazados de Windows y Linux de la UD4

Los resultados se entregarán en un fichero con capturas de pantalla de todos los pasos. Las imágenes deben mostrar claramente la MV del autor.

### A. Windows 7-10-UD4

Utilizando la herramienta para programación de tareas de Windows 7 configura la realización de una copia de seguridad de la carpeta de documentos. Para ello usar como herramienta de backups **Cobian Backup** que se ejecutará a una determinada hora.

A continuación con el programador de tareas de Windows 7 configura una tarea que suponga que cuando se inicie Windows se lance el Internet Explorer.

Programar la actualización del sistema (Windows Update) todos los lunes a las 9:00 horas

Finalmente programar la tarea de apagado del equipo a una determinada hora.

### B. Ubuntu

En modo consola y obteniendo información de Internet ([enlace](#)) configura las tareas siguientes:

- Apagar el equipo a una hora determinada por ejemplo todos los miércoles a las 13:45
- Actualizar el sistema operativo los martes a las 10.50 y los miércoles a las 10:30
- Crea una tarea para que todos los días a las 12 horas borre **el contenido** del directorio */home/usuario/tmp*
- Utilizando el comando **at** programa una sola tarea para que a las 12:40 te salga un mensaje por pantalla: “Hola Bienvenido”. Este mensaje estará contenido en un fichero llamado ***hola.txt*** ¿qué diferencia hay entre **crontab** y **at**?

## Avanzado de unidades posteriores sobre Linux

### Permisos sobre archivos (ver UD4)

En modo consola crea un fichero llamando **pruebas.txt**. Escribe los comandos y las capturas de pantalla correspondientes (son 21) donde se reflejen los cambios de los permisos, **indicando en cada caso que tipo de permisos** tiene el **propietario** del archivo, el **grupo** del propietario y los otros **usuarios**. Por ejemplo,

```
#chmod 777 pruebas && ls -al pruebas
```

**Explicación:** todos los usuarios tienen todos los permisos de lectura, escritura y ejecución

---

### Introducción a los comandos de linux.

Poco a poco iremos introduciendo los comandos en Linux para adelantar su estudio en el 3º parcial.

Deberás ejecutar los siguientes comandos, hacer una captura de pantalla y explicar su uso según el formato indicado anteriormente. En algunos de ellos deberás listar para ver el resultado (**ls -al**)

*#comando*

### Explicación del comando.

- |                          |                             |                          |
|--------------------------|-----------------------------|--------------------------|
| – cal 1969               | – uname -a                  | – uptime                 |
| – pwd                    | – ls /etc >> <i>fichero</i> | – ls -al /etc/s*         |
| – history                | – cat fichero               | – wc <i>fichero</i>      |
| – whereis <i>archivo</i> | – passwd <i>usuario</i>     | – head -5 <i>fichero</i> |
| – clear                  | – wall                      | – tail -5 <i>fichero</i> |
| – who                    | – ifconfig                  | – df -h                  |