



Crontab

Programación de Tareas con Crontab

Objetivo: Aprender a programar tareas automáticas con `cron` y `crontab` en sistemas Linux.

1. Introducción a Cron y Crontab

- ¿Qué es `cron` ?
 - Servicio en Linux para la ejecución automática de comandos en un horario definido.
 - Casos de uso: copias de seguridad, limpieza de logs, ejecución de scripts.
- ¿Qué es `crontab` ?
 - Archivo donde los usuarios pueden definir sus propias tareas programadas.
 - Diferencia entre `cron` (el servicio) y `crontab` (el archivo de configuración del usuario).

2. Sintaxis y Formato de Crontab

Formato básico de una línea en `crontab -e`

MIN HORA DIA MES DIA-SEMANA COMANDO

Campo	Valores posibles	Ejemplo
Minuto	0-59	30
Hora	0-23	14 (2 PM)

Día	1-31	5 (Día 5 del mes)
Mes	1-12	7 (Julio)
Día Semana	0-6 (Dom = 0)	1 (Lunes)

Ejemplos de programaciones comunes

1. Ejecutar un script todos los días a las 3 AM

```
0 3 * * * /home/usuario/backup.sh
```

2. Ejecutar cada lunes a las 7:30 AM

```
30 7 * * 1 /home/usuario/script.sh
```

3. Ejecutar cada 15 minutos

```
*/15 * * * * /home/usuario/logrotate.sh
```

4. Ejecutar cada primer día del mes a medianoche

```
0 0 1 * * /home/usuario/mensual.sh
```

Caracteres especiales

Caracter	Significado
*	Cualquier valor
,	Lista de valores (ej: 1,15,30)
-	Rango de valores (ej: 1-5 ejecuta del día 1 al 5)
/	Incremento (ej: */10 cada 10 minutos)

3. Comandos útiles en Crontab

- Listar **crontabs** del usuario actual

```
crontab -l
```

- Editar crontab del usuario actual

```
crontab -e
```

- **Eliminar el crontab del usuario actual**

```
crontab -r
```

- **Ejecutar crontab para otro usuario (root necesario)**

```
crontab -u usuario -l
```

- **Ver logs de `cron` para depuración**

```
cat /var/log/syslog | grep cron
```

4. Variables de Entorno y Redirección de Salida

- **Variables comunes en crontab**

- `SHELL=/bin/bash` → Define el shell a usar.
- `PATH=/usr/bin:/bin:/usr/local/bin` → Evita errores por rutas incompletas.
- `MAILTO=usuario@ejemplo.com` → Enviar salida de cron por correo.

- **Redirección de salida**

- Guardar salida en un archivo:

```
* * * * * /home/usuario/script.sh >> /home/usuario/salida.log 2>&1
```

- Silenciar salida:

```
* * * * * /home/usuario/script.sh > /dev/null 2>&1
```

5. Ejercicios Prácticos

1. **Ejercicio 1:** Crea una tarea en `crontab` para mostrar un mensaje cada minuto en un archivo:

```
* * * * * echo "Hola Educación IT, crontab funciona" >> ~/mensaje.txt
```

2. **Ejercicio 2:** Programa un script que se ejecute todos los viernes a las 10 PM y haga un respaldo de una carpeta (`/home/usuario/documentos`).
 3. **Ejercicio 3:** Usa `crontab -l` para verificar tus tareas programadas.
 4. **Ejercicio 4:** Configura `MAILTO` en `crontab` para recibir notificaciones cuando una tarea falle.
 5. **Ejercicio 5:** Crea una tarea que se ejecute solo en los días hábiles (lunes a viernes) a las 9 AM.
-

6. Información Util para tener en cuenta

- **Errores comunes**
 - Usar rutas relativas en scripts.
 - Falta de permisos de ejecución en scripts.
 - Problemas con la variable `PATH` .
 - Tareas no ejecutadas por falta de permisos.
-

Materiales Adicionales

- Documentación oficial: `man crontab`
- Tutorial interactivo: crontab.guru



Carpeta de Crontab