



# cron

## Automatización de tareas

### Laboratorio de Sistemas Operativos II



La **automatización** consiste en usar la tecnología disponible para realizar tareas con mínima intervención humana.  
Se implementa donde se realizan tareas repetitivas.



- Reducción de las tareas manuales
- Optimización del tiempo
- Confiabilidad y eficiencia
- Mejoras en la gestión

# ¿Cuáles tareas se automatizan?

- Actividades de gestión periódicas y control del sistema.
- Limpieza periódica de archivos temporales (/tmp)
- Copias de seguridad.
- Generación de informes regulares del sistema.
- Programar el apagado automático del equipo.
- Programar tareas periódicas desde las necesidades del usuario (recordatorios, o la descarga programada de información en horas de poco tráfico).

## cron

GNU/Linux cuenta con el servicio o demonio **cron** que permite a los usuarios ejecutar automáticamente comandos o scripts (tareas) programadas a una hora o fecha específica, o a intervalos regulares de tiempo.



# cron

El nombre cron viene del griego **chronos** que significa "tiempo".



# Configuración de la zona horaria en el sistema

- La ejecución de Cron se realiza de acuerdo a la hora establecida en el sistema operativo.
- La hora y zona horaria deben estar correctamente configuradas bajo servidores NTP (Network Time Protocol) o Protocolo de tiempo de red. .
- Su función principal es la de sincronizar los relojes de los sistemas informáticos para evitar diferencias horarias.
- Comando relacionado: **timedatectl**

# timedatectl

- Ver zona horaria actual del sistema y si está sincronizado con un servidor NTP:

```
#timedatectl
```

- Ver zonas horarias disponibles para Argentina:

```
#timedatectl list-timezones | grep America/Argentina
```

- Establecer zona horaria:

```
#timedatectl set-timezone America/Argentina/Buenos_Aires
```

# ¿Cómo funciona cron?

- **Cron** es un programa que administra procesos en segundo plano (**demonio**).
- Es iniciado cada vez que el sistema se inicia y se ejecuta en background.
- Revisa cada minuto el archivo que contiene la tabla de tareas (**crontab**) en búsqueda de tareas que se deban cumplir y las ejecuta.

# Repaso de conceptos

<b>cron</b>	Es el servicio o demonio que ejecuta las tareas programadas.
<b>crontab</b>	Es el archivo de configuración en el cual se definen las tareas programadas.
<b>tarea cron</b>	Es lo que se va a ejecutar y está definido en crontab.

# Administrar el servicio cron

Operacion	Comando
Iniciar el servicio	<code># systemctl start cron</code>
Detener el servicio	<code># systemctl stop cron</code>
Reiniciar el servicio	<code># systemctl restart cron</code>
Ver el estado del servicio	<code># systemctl status cron</code>



# Crontabs de usuario

- Comando: **crontab -e**
  - Abre crontab de usuario conectado
- Archivos de texto que gestionan la programación de los trabajos cron definidos por un usuario.
- Su nombre es igual al usuario que los creó
- Su ubicación es (depende de la distro) :  
**/var/spool/cron/crontabs**
- Cada línea contiene **seis** campos separados por un espacio

# Formato de línea de crontab de usuario

Minuto Hora DiaDelMes Mes DiaDeLaSemana Comando

Comando, script o programa

Día en la semana en que se ejecutará el comando, puede ser de 0 a 7, o por el nombre del día en inglés, solo las tres primeras letras. (0 y 7 = domingo)

Mes en que el comando se ejecutará, puede ser indicado del 1 al 12), o por el nombre del mes en inglés, solo las tres primeras letras.

Día del mes en que se quiere ejecutar el comando. Va del 1 al 31.

Hora en que el comando será ejecutado, en un formato de 24 horas, los valores deben estar entre 0 y 23, 0 es medianoche.

Corresponde al minuto en que se va a ejecutar el script, el valor va de 0 a 59

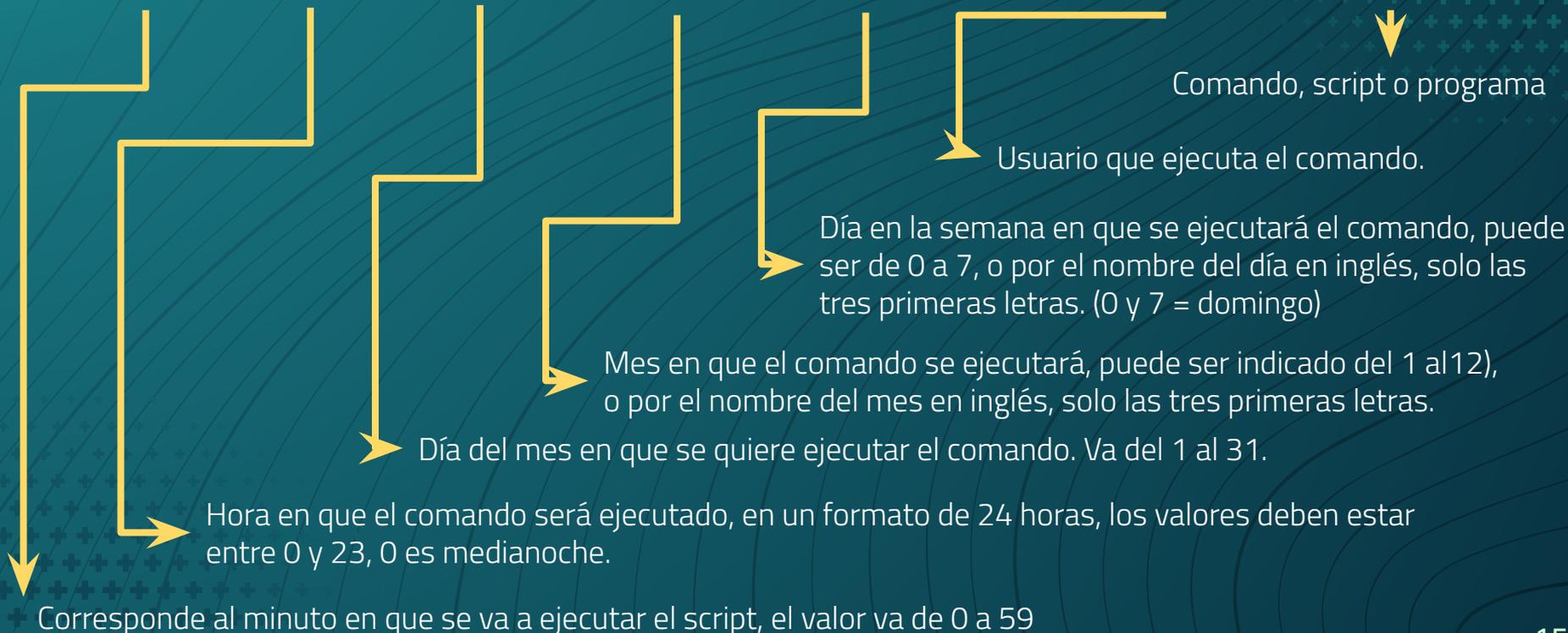
# Crontab del sistema



- Los crontabs de sistema son archivos de texto que gestionan la programación de los trabajos del cron del sistema y sólo pueden ser editados por el usuario root con un editor de texto.
- El archivo **/etc/crontab** y todos los que se encuentran del directorio `/etc/cron.d` son crontabs del sistema.
- La mayoría de las distribuciones incluyen los directorios `/etc/cron.hourly`, `/etc/cron.daily`, `/etc/cron.weekly` y `/etc/cron.monthly` para frecuencia de tareas específica.
- Cada Línea contiene **siete** campos para cada tarea programada.

# Formato de línea de contrab

Minuto Hora DiaDelMes Mes DiaDeLaSemana **Usuario** Comando



# Detalle del formato

<b>Minuto</b>	Controla el minuto de la hora en que el comando será ejecutado, este valor debe de estar entre 0 y 59.
<b>Hora</b>	Controla la hora en que el comando será ejecutado, se especifica en un formato de 24 horas, los valores deben estar entre 0 y 23, 0 es medianoche.
<b>Día del Mes</b>	Día del mes en que se quiere ejecutar el comando. Por ejemplo se indicaría 20, para ejecutar el comando el día 20 del mes.
<b>Mes</b>	Mes en que el comando se ejecutará, puede ser indicado numéricamente (1-12), o por el nombre del mes en inglés, solo las tres primeras letras.
<b>Día de la semana</b>	Día en la semana en que se ejecutará el comando, puede ser numérico (0-7) o por el nombre del día en inglés, solo las tres primeras letras. (0 y 7 = domingo)
<b>Usuario</b>	Usuario que ejecuta el comando.
<b>Comando</b>	Comando, script o programa que se desea ejecutar. Este campo puede contener múltiples palabras y espacios.

Un asterisco \* como valor en los primeros cinco campos, indicará inicio-fin del campo, es decir todo. Un \* en el campo de minuto indicará todos los minutos.

# Horarios predefinidos

Entrada	Descripción	Equivale a:
@yearly	Se ejecuta una vez al año	0 0 1 1 *
@monthly	Se ejecuta una vez al mes	0 0 1 * *
@weekly	Se ejecuta una vez a la semana	0 0 * * 0
@daily	Se ejecuta una vez al día	0 0 * * *
@hourly	Se ejecuta una vez cada hora	0 * * * *

# Cómo editar crontab

- Como root, se editará el /etc/crontab directamente:

```
$sudo crontab -e o #crontab -e
```

- El crontab de root permite programar tareas que requieren privilegios de superusuario, como modificaciones al sistema o actualizaciones de paquetes.
- Al editar el crontab de root se debe tener cuidado, cualquier error puede afectar al sistema.
- Como usuario normal, se utiliza el comando crontab para que de este modo cada usuario obtenga automáticamente su propio crontab en /var/spool/cron/crontabs

```
$crontab -e
```

# Otras opciones del comando crontab

- Lista el crontab actual del usuario, sus tareas de cron

```
$crontab -l
```

- Elimina el crontab actual del usuario (cuidado si eres root!!)

```
$crontab -r
```

## Ejemplos:

- Ejecutará el script actualizar.sh a las 10:15 a.m. todos los días

```
15 10 * * * usuario /home/usuario/scripts/actualizar.sh
```

- El día 21 de noviembre a las 6:30 el usuario correrá el script backup.sh

```
30 6 21 11 * usuario /home/usuario/scripts/backup.sh
```

- A las 12:00 del día todos los días primero, quince y 28 de cada mes

```
00 12 1,15,28 * * root /opt/generarNominas.sh
```

## Más Ejemplos:

- Cada quince minutos de las 11:00a.m. a las 14:00p.m.

```
*/15 11-14 * * * root /opt/checkDNS.sh
```

- Cada 6 horas de los días 1 al 10 de cada mes y que el día sea miércoles

```
0 */6 1-10 * 3 root /opt/checkDNS.sh
```

- root ejecutará una actualización los domingos (sun) a las 10:45 a.m

```
45 10 * * sun root apt-get -y update
```

# Cron por hora, diario, semanal, mensual

- **/etc/cron.daily**

Cron ejecuta todos los scripts ubicados en cron.daily una vez al día en la hora especificada en /etc/crontab.

- **/etc/cron.hourly**

Los scripts contenidos en el directorio /etc/cron.hourly se ejecutarán cada hora. ( 24 veces al día si el equipo está siempre encendido). La hora de ejecución se especifica en /etc/crontab.

- **/etc/cron.weekly**

Cron ejecuta todos los scripts ubicados en cron.weekly una vez a la semana en el día y hora especificados en /etc/crontab.

- **/etc/cron.monthly**

Cron ejecuta todos los scripts ubicados en cron.monthly mensualmente en el día y hora especificados en /etc/crontab



# Fin.-

¿Preguntas?

Fuentes: <https://learning.lpi.org/es/>