

[VPS](#) Mar 22, 2023 Diego V. 5min de lectura

Qué es la sintaxis Crontab: Crontab en Linux y ejemplos útiles



La automatización es uno de los aspectos clave de cualquier sistema, ya sea un servidor físico o un [servidor privado virtual \(VPS\)](#). Si la automatización se configura correctamente, puedes ahorrar cientos de horas preciosas para el usuario y hacer que el flujo de trabajo general sea mucho más eficiente. Una de las herramientas clave para programar tareas es **cron**.

En este tutorial, vamos a cubrir la sintaxis Crontab, veremos un resumen de las diferencias entre cron job, cron y crontab, y te mostraremos algunos casos de uso útiles cron jobs para un sistema operativo basado en Linux.

Esta hoja de trucos hará que los comandos de Linux sean fáciles de usar

Obtener la hoja ya



Tabla de Contenidos

- [¿Qué es Crontab y Cron Job? >](#)
- [¿Qué es la sintaxis Crontab? >](#)
- [Formato y valores de Crontab >](#)
- [Comandos Crontab >](#)
- [Cómo utilizar Crontab ejemplos de sintaxis Crontab >](#)
 - [Programar un trabajo para una hora determinada >](#)
 - [Ver entradas Crontab >](#)
 - [Editar entradas Crontab >](#)
 - [Programar un trabajo para cada minuto >](#)
 - [Programar un trabajo de fondo cada día >](#)
 - [Programar un trabajo para un intervalo de tiempo determinado >](#)
 - [Programar un Cron Job al principio de cada mes >](#)
 - [Programar un trabajo para más de una vez >](#)
 - [Ejecutar un comando Linux después de cada reinicio >](#)
 - [¿Dónde se encuentra el archivo Crontab? >](#)
- [Acciones adicionales de Crontab >](#)
 - [Crear un archivo de registro >](#)
 - [Desactivar correo electrónico >](#)
 - [Entorno Crontab >](#)

¿Qué es Crontab y Cron Job?

En cuanto a los cron jobs, cabe destacar tres términos:

Cron daemon (crond) o simplemente **cron** es un ejecutable que permite a los usuarios realizar tareas y ejecutarlas automáticamente a una hora determinada.

Cron job es cualquier tarea que un usuario programe utilizando [cron](#).

¿Qué es la sintaxis Crontab?

La **tabla cron** o **Crontab** es un archivo que contiene todas las programaciones de las tareas cron que un usuario desea ejecutar regularmente. Los comandos se escriben uno por línea y ordenan al cron daemon que ejecute una tarea a una hora determinada.

Formato y valores de Crontab

Para que el cron daemon comprenda correctamente las instrucciones, debe utilizarse la sintaxis crontab correcta. La sintaxis de crontab consta de cinco campos. Cada uno de ellos puede rellenarse con cualquiera de los valores que se muestran en la siguiente tabla:

Campo	Valores posibles

Hora	0-23
Día del mes	1-31
Mes	1-12
Día de la semana	0-6. 0 representa el domingo. En algunos sistemas, el valor 7 representa el domingo.
Comando	Orden a ejecutar

Aparte de los valores de crontab posibles, es necesario memorizar algunos caracteres especiales:

Símbolo	Significado	Ejemplo
* (asterisco)	Seleccionar todos los valores posibles de un campo	Coloca * en el campo hora para ejecutar la tarea cada hora
, (coma)	Se utiliza una coma para separar varios valores	0,3,5 en el campo día de la semana hará que la tarea se ejecute el domingo y el miércoles
– (guión)	Permite fijar un intervalo de valores	10-15 en el campo día del mes ejecutará la tarea desde el día 10 al 15 del mes
/ (separador)	Se utiliza un separador para dividir los valores	*/10 en el campo hora hará que la tarea se ejecute cada 10 horas
L	Se utiliza en los campos de día del mes o día de la semana	1L en el campo día de la semana ejecutará la tarea el último lunes de un mes determinado
W	W se utiliza para determinar el día de la semana más próximo	0W en el campo día del mes hará que la tarea se ejecute el domingo más cercano de un mes determinado
# (numeral)	Se utiliza para determinar el día de la semana	2#3 en el campo día del mes hará que la tarea se ejecute el tercer martes del mes
? (signo de interrogación)	Se utiliza en los campos de día del mes y día de la semana	? en el campo del día del mes se leerá como sin valor específico



¡Importante! Ten en cuenta que todos los campos deben rellenarse al crear una tarea cron. Si

Comandos Crontab

Sólo se utilizan unos pocos comandos cron en la línea de comandos, lo que facilita la creación, eliminación y gestión de entradas cron:

- **crontab -e**: se utiliza para editar crontabs del sistema. Este comando creará un nuevo crontab si aún no se ha hecho.
- **crontab -l**: se utiliza para ver las entradas crontab (trabajos cron) y mostrar el contenido del archivo crontab del sistema.
- **crontab -r**: eliminará el archivo crontab actual.
- **crontab -i**: mostrará un aviso antes de eliminar el crontab de un usuario. Se recomienda utilizarlo junto con el indicador **-r**, convirtiéndolo en el indicador **-ri**.

Cómo utilizar Crontab: ejemplos de sintaxis Crontab

En primer lugar, utiliza el comando crontab para crear tu primera entrada crontab:

Plain text
Copy to clipboard
Open code in new window
EnlighterJS 3 Syntax Highlighter
crontab -e
crontab -e

```
crontab -e
```

Te pedirá que elijas un editor de texto. Te recomendamos usar **nano**, la primera opción en nuestro ejemplo:

Después, serás dirigido al archivo crontab. Para añadir nuevas entradas, simplemente elige una nueva línea y continúa con el cronjob.



¡Importante! Ten en cuenta que crontab utiliza la fecha y hora actuales del sistema, lo que significa que la zona horaria se comparte con el sistema raíz.

Programar un trabajo para una hora determinada

Una de las formas más sencillas de probar cron es programar una tarea para una hora determinada. Por ejemplo, la siguiente tarea cron ejecutará nuestro **script .sh** a la hora especificada, que es el **15 de agosto a las 15:30**.

Plain text
Copy to clipboard
Open code in new window
EnlighterJS 3 Syntax Highlighter

Ver entradas Crontab

Dado que todas las tareas cron se almacenan en un archivo crontab, puedes ver las que ya se están ejecutando. Para ello, utiliza el siguiente comando para mostrar el contenido de tus archivos crontab:

Plain text
Copy to clipboard
Open code in new window
EnlighterJS 3 Syntax Highlighter
crontab -l
crontab -l

```
crontab -l
```

Editar entradas Crontab

Para modificar entradas crontab ya existentes, utiliza este comando:

Plain text
Copy to clipboard
Open code in new window
EnlighterJS 3 Syntax Highlighter
crontab -e
crontab -e

```
crontab -e
```

Programar un trabajo para cada minuto

No supone ningún esfuerzo programar tareas para que se ejecuten cada minuto. En el siguiente ejemplo, ordenaremos la ejecución del comando cat para que se ejecute periódicamente:

Plain text
Copy to clipboard
Open code in new window
EnlighterJS 3 Syntax Highlighter
***** cat /home/helloworld.sh
***** cat /home/helloworld.sh

```
* * * * * cat /home/helloworld.sh
```

Programar un trabajo de fondo cada día

Para programar una tarea en segundo plano que se ejecute todos los días, puedes utilizar el comando cron **@daily**:

Plain text
Copy to clipboard
Open code in new window
EnlighterJS 3 Syntax Highlighter
@daily cat /home/helloworld.sh
@daily cat /home/helloworld.sh

Ten en cuenta que se va a ejecutar el script todos los días a las **12 de la mañana**.

Programar un trabajo para un intervalo de tiempo determinado

Es posible programar un trabajo para un intervalo de tiempo determinado. Por ejemplo, todos los días de la semana, incluidos los fines de semana, de **8.00** a **17.00**. El resultado final sería el siguiente:

Plain text

Copy to clipboard

Open code in new window

EnlighterJS 3 Syntax Highlighter

```
00 08-17 * * * cat /home/helloworld.sh
```

```
00 08-17 * * * cat /home/helloworld.sh
```

```
00 08-17 * * * cat /home/helloworld.sh
```

Aquí hay otro ejemplo del mismo cron, pero sólo los fines de semana:

Plain text

Copy to clipboard

Open code in new window

EnlighterJS 3 Syntax Highlighter

```
00 08-17 * * 6-0 cat /home/helloworld.sh
```

```
00 08-17 * * 6-0 cat /home/helloworld.sh
```

```
00 08-17 * * 6-0 cat /home/helloworld.sh
```

Programar un Cron Job al principio de cada mes

Para programar un trabajo al principio de cada mes, puedes utilizar el operador **@monthly**:

Plain text

Copy to clipboard

Open code in new window

EnlighterJS 3 Syntax Highlighter

```
@monthly cat /home/helloworld.sh
```

```
@monthly cat /home/helloworld.sh
```

```
@monthly cat /home/helloworld.sh
```

Ten en cuenta que esto ejecutará el trabajo a **las 12am** del primer día de cada mes. Del mismo modo, existe un operador **@yearly** que ejecutará el trabajo el **primer día** de cada año.

Programar un trabajo para más de una vez

Los usuarios pueden programar una tarea cron para que se ejecute más de una vez, por ejemplo, cinco veces al día. En el siguiente ejemplo, configuraremos un trabajo para que se ejecute a las **12pm**, **3pm**, **5pm**, **7pm** y **9pm**.

Plain text

Copy to clipboard

Open code in new window

EnlighterJS 3 Syntax Highlighter

```
0 12,15,17,19,21 * * * cat /home/helloworld.sh
```

```
0 12,15,17,19,21 * * * cat /home/helloworld.sh
```

```
0 12,15,17,19,21 * * * cat /home/helloworld.sh
```

Ejecutar un comando Linux después de cada reinicio

De forma similar a los operadores **@daily**, **@monthly** y **@yearly**, bash también viene con un comando **@reboot**. Los usuarios pueden utilizarlo para programar una tarea que se ejecutará cada vez que el sistema se reinicie:

Plain text

Copy to clipboard

Open code in new window

EnlighterJS 3 Syntax Highlighter

```
@reboot cat /home/helloworld.sh
```

```
@reboot cat /home/helloworld.sh
```

```
@reboot cat /home/helloworld.sh
```

- **MacOS:** `/var/at/tabs`
- **Sistemas basados en Debian:** `/var/spool/cron/crontabs/`
- **Sistemas basados en Red Hat:** `/var/spool/cron`

Acciones adicionales de Crontab

Aparte de la funcionalidad actual, crontab también es capaz de realizar acciones adicionales como crear un registro de ejecución cron o desactivar las notificaciones por correo electrónico. Consulta las secciones siguientes para obtener más información.

Crear un archivo de registro

La forma más sencilla de registrar todas las salidas de los trabajos cron es utilizar la siguiente lógica:

Plain text

Copy to clipboard

Open code in new window

EnlighterJS 3 Syntax Highlighter

```
**** cat /home/helloworld.sh >> /var/log/logs.log 2>&1
**** cat /home/helloworld.sh >> /var/log/logs.log 2>&1
```

```
* * * * * cat /home/helloworld.sh >> /var/log/logs.log 2>&1
```

Este comando guardará todas las salidas de los trabajos cron en un archivo llamado **logs.log**

Desactivar correo electrónico

Dado que cron envía un correo electrónico al usuario después de cada trabajo, desactivar esta funcionalidad puede ser beneficioso para evitar el spam. Sólo tienes que añadir esta línea al final de tu trabajo cron:

Plain text

Copy to clipboard

Open code in new window

EnlighterJS 3 Syntax Highlighter

```
>/dev/null 2>&1
>/dev/null 2>&1
```

```
>/dev/null 2>&1
```

Entorno Crontab

Crontab permite a sus usuarios definir variables de entorno. Esto puede hacerse con el comando **crontab -e** antes mencionado. Al definir variables, consulta la lista siguiente:

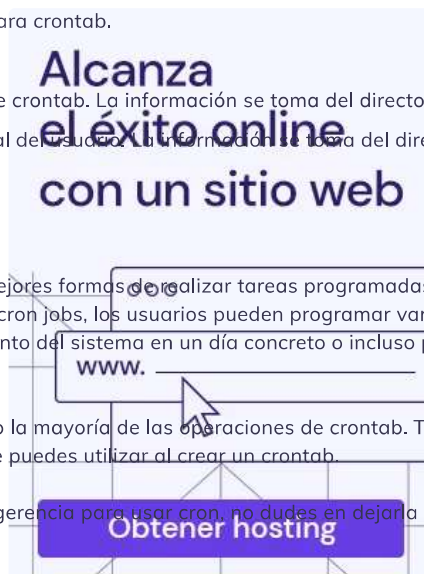
- **PATH:** ruta por defecto para crontab.
- **SHELL:** shell por defecto.
- **LOGNAME:** propietario de crontab. La información se toma del directorio `/etc/passwd`.
- **HOME:** directorio personal del usuario. La información se toma del directorio `/etc/passwd`.

Conclusión

Los cron jobs son una de las mejores formas de realizar tareas programadas para **instancias virtuales** y sistemas Linux físicos. Con los cron jobs, los usuarios pueden programar varias tareas para su sistema. Por ejemplo, realizar el mantenimiento del sistema en un día concreto o incluso programarlo para que se ejecute todos los días de la semana.

En este tutorial, hemos cubierto la mayoría de las operaciones de crontab. También hemos proporcionado algunos ejemplos prácticos que puedes utilizar al crear un crontab.

Si tienes alguna pregunta o sugerencia para usar cron, no dudes en dejarla en la sección de comentarios.



Tutoriales relacionados

20 Mar • [VPS](#)

[Cómo configurar un servidor de correo propio en 2024](#)

Los empresarios suelen utilizar servidores de correo preconfigurados de terceros para alojar sus servicios de correo. Aunque son suficientes para...

[Por Diana Catalina Herrera Infante](#)

16 Feb • [VPS](#)

[Los 10 mejores proveedores de hosting VPS para 2024](#)

Con un hardware de alto rendimiento y flexibilidad, un servidor privado virtual (VPS) es excelente para alojar sitios o aplicaciones web que consumen...

[Por Diana Catalina Herrera Infante](#)

16 Feb • [VPS](#)

[Cómo alojar un bot de Discord: creación y desarrollo de tu propio bot en un VPS en 2024](#)

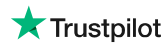
Añadir bots a tu servidor de Discord te permite automatizar tareas y añadir nuevas funciones. Si bien hay muchos bots Discord populares que están...

Lo que dicen nuestros clientes

Excelente



En base a 24.958 opiniones



Deja una respuesta

Comentar*

Nombre*

Email*

☐ Al utilizar este formulario, aceptas que tus datos personales serán procesados de acuerdo con nuestra [Política de privacidad](#).

Enviar



Somos un proveedor de hosting web con la misión de llevar el éxito a todos los que están en Internet. Lo hacemos mejorando constantemente la tecnología del servidor, brindando soporte profesional y haciendo que la experiencia de hosting web sea perfecta.



[Y Más](#)

HOSTING

[Hosting web](#)

[Hosting para profesionales](#)

Servicios

[Cloud hosting](#)

[Cheap WordPress Hosting](#)

[Correo corporativo](#)

[Hosting CMS](#)

[Hosting eCommerce](#)

[Hosting gratuito](#)

[Tienda online](#)

[Creador de páginas web](#)

[Creador de logos](#)

[Generador de nombres para empresas](#)

DOMINIOS

[Comprar dominio](#)

[Transferir dominio](#)

[Dominio gratis](#)

[Dominio .xyz](#)

[Dominios baratos](#)

[Extensiones de Dominio](#)

[WHOIS](#)

[Certificado SSL gratis](#)

AYUDA

[Tutoriales](#)

[Base de Conocimientos](#)

[Reportar abuso](#)

INFORMACIÓN

[Migrar a Hostinger](#)

[Estado del sistema](#)

[Programa de afiliados](#)

[Formas de pago](#)

[Muro de la fama](#)

[Opiniones](#)

[Precios](#)

[Mapa del sitio](#)

EMPRESA

[Acerca de Hostinger](#)

[Tecnología](#)

[Mapa del sitio \(inglés\)](#)

[Contáctanos](#)

[Blog](#)

LEGAL

[Política de privacidad](#)

[Términos de servicio](#)

