



# Programación de Sistemas

## CCPG1008

---

Federico Domínguez, PhD.

Unidad 5 – Sesión 2: Instalación desde código fuente en Linux

# Contenido

---

Introducción a la instalación de software desde código Fuente

Práctica: Compilación, instalación y prueba de un servidor web

En algunos casos, especialmente cuando se desea instalar software experimental o la última versión de alguna herramienta, es necesario instalar software desde el código fuente.

---

Distribuciones como Ubuntu o Centos tienen una enorme base de paquetes de software y rara vez es necesario instalar software desde el código fuente.

Es algo común en el ámbito de sistemas embebidos debido a la naturaleza experimental de las plataformas a usar o la ventaja de poder compilar para la plataforma de hardware a usar.

# Antes de empezar, instalar herramientas de compilación.

---

- *gcc* y otros
- *Make*
- *CMake*
- Autoconf
- *git*, *wget*, ...
- Las herramientas de compilación están empaquetadas convenientemente en un solo paquete en Ubuntu:
  - `sudo apt install build-essential`

# Típicamente el código fuente es distribuido en un *tarball* o a través de repositorios.

---

*Tar* es una herramienta para archivar ficheros en un solo archivo *.tar*. Mantiene parámetros como permisos, directorios y otros. Un archivo *.tar* es usado típicamente como respaldo o para distribución, es conocido como un *tarball*.

*gzip* y *bzip* son herramientas de compresión, típicamente un *tarball* es además comprimido con una de estas herramientas.

Para descomprimir y expandir los contenidos de un *tarball* comprimido:

- `tar -xzvf file.tar.gz`
- `tar -xjvf file.tar.bz2`

# Para incrementar la portabilidad del código fuente se usa la herramienta *Autoconf*.

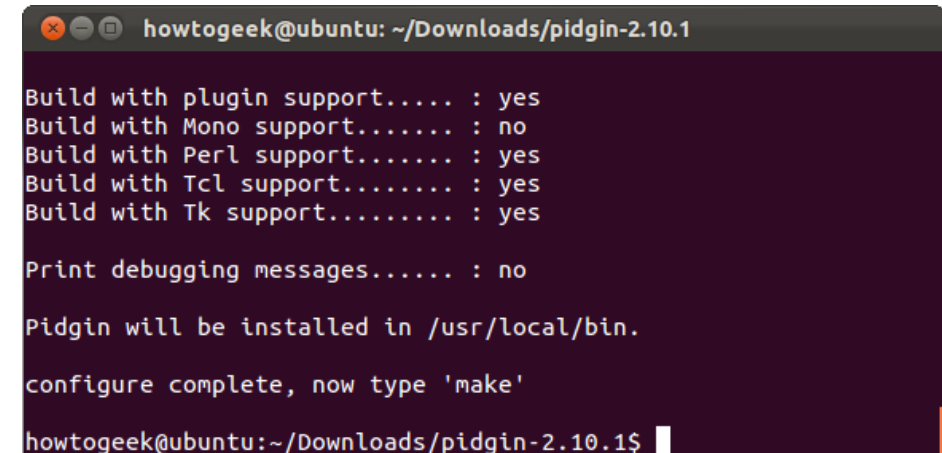
---

Aunque no siempre es el caso, se debe leer antes el archivo *README* o *INSTALL* en la carpeta del código fuente. Típicamente, para empezar se debe ejecutar el script:

- `./configure`

Este script revisa las librerías presentes y el ambiente de compilación para luego generar el archivo *Makefile* apropiado.

Es común que este paso falle inicialmente y sea necesario instalar más dependencias.



```
howtogeek@ubuntu: ~/Downloads/pidgin-2.10.1
Build with plugin support..... : yes
Build with Mono support..... : no
Build with Perl support..... : yes
Build with Tcl support..... : yes
Build with Tk support..... : yes

Print debugging messages..... : no

Pidgin will be installed in /usr/local/bin.

configure complete, now type 'make'

howtogeek@ubuntu:~/Downloads/pidgin-2.10.1$
```

# Finalmente es necesario compilar e instalar con make.

---

Una vez generado el archivo *Makefile* se puede compilar:

- `make`

Este paso puede tardar desde algunos segundos hasta días dependiendo de los recursos computacionales del host y la complejidad del código fuente.

Para instalar, el archivo *Makefile* tiene un *target* “*install*”, el cuál normalmente instala los ejecutables en */usr/sbin* o */usr/local/sbin* y las librerías en */usr/lib*. Por este motivo, es necesario ejecutar este comando con permisos de *root*.

- `sudo make install`

Si no se ha borrado la carpeta original con código fuente, el software puede ser desinstalado con:

- `sudo make uninstall`

# Referencia

---

<https://www.howtogeek.com/105413/how-to-compile-and-install-from-source-on-ubuntu/>