

# tldr pages book

Simplified and community-driven man pages

*Generated on Mon Oct 14 19:33:49 2024*

Website: <https://tldr.sh>

GitHub: <https://github.com/tldr-pages/tldr>

# Android

# am

Administrador de actividades de Android.

Más información: <https://developer.android.com/tools/adb#am>.

- Inicia una actividad específica:

```
am start -n {{com.android.settings/.Settings}}
```

- Inicia una actividad y le suministra datos:

```
am start -a {{android.intent.action.VIEW}} -d {{tel:123}}
```

- Inicia una actividad que coincide con una acción y categoría específicas:

```
am start -a {{android.intent.action.MAIN}} -c  
{{android.intent.category.HOME}}
```

- Convierte una intención en una URI:

```
am to-uri -a {{android.intent.action.VIEW}} -d {{tel:123}}
```

# bugreport

Muestra un informe de error de Android.

Este comando solo se puede usar a través de **adb shell**.

Más información: <https://cs.android.com/android/platform/superproject/+/main:frameworks/native/cmds/bugreport>.

- Muestra un informe completo de errores de un dispositivo Android:

```
bugreport
```

# bugreportz

Genera un informe comprimido de errores de Android.

Este comando solo se puede usar a través de **adb shell**.

Más información: <https://cs.android.com/android/platform/superproject/+/main:frameworks/native/cmds/bugreportz>.

- Genera un informe completo de errores comprimido de un dispositivo Android:

```
bugreportz
```

- Muestra el progreso de una operación bugreportz en ejecución:

```
bugreportz -p
```

- Muestra ayuda:

```
bugreportz -h
```

- Muestra la versión de bugreportz:

```
bugreportz -v
```

# cmd

Administrador de servicios Android.

Más información: <https://cs.android.com/android/platform/superproject/+/main:frameworks/native/cmds/cmd/>.

- Enumera todos los servicios en ejecución:

```
cmd -l
```

- Llama a un servicio específico:

```
cmd {{alarm}}
```

- Llama a un servicio con argumentos:

```
cmd {{vibrator}} {{vibrate 300}}
```

# dalvikvm

Máquina virtual Java en Android.

Más información: <https://source.android.com/docs/core/runtime>.

- Inicia un programa Java:

```
dalvikvm -classpath {{ruta/al/archivo.jar}} {{classname}}
```

# dumpsys

Suministra información sobre los servicios del sistema Android.

Este comando solo se puede usar a través de **adb shell**.

Más información: <https://developer.android.com/tools/dumpsys>.

- Obtén los resultados de diagnóstico para todos los servicios del sistema:

```
dumpsys
```

- Obtén los resultados de diagnóstico para un servicio de sistema específico:

```
dumpsys {{service}}
```

- Enumera todos los servicios que **dumpsys** sobre los que puede proporcionar información:

```
dumpsys -l
```

- Enumera los argumentos específicos del servicio para un servicio determinado:

```
dumpsys {{service}} -h
```

- Excluye un servicio específico de la salida de diagnóstico:

```
dumpsys --skip {{service}}
```

- Especifica un período de tiempo de espera en segundos (predeterminado en 10 segundos):

```
dumpsys -t {{seconds}}
```



# getprop

Muestra información sobre las propiedades del sistema Android.

Más información: <https://manned.org/getprop>.

- Muestra información sobre las propiedades del sistema Android:

```
getprop
```

- Muestra información sobre una propiedad específica:

```
getprop {{prop}}
```

- Muestra el nivel SDK de la API:

```
getprop {{ro.build.version.sdk}}
```

- Muestra la versión de Android instalada:

```
getprop {{ro.build.version.release}}
```

- Muestra el modelo del dispositivo Android:

```
getprop {{ro.vendor.product.model}}
```

- Muestra el estado de desbloqueo del OEM:

```
getprop {{ro.oem_unlock_supported}}
```

- Muestra la dirección MAC de la tarjeta Wi-Fi de Android:

```
getprop {{ro.boot.wifimacaddr}}
```

# input

Envía códigos de eventos o gestos de pantalla táctil a un dispositivo Android.

Este comando solo se puede usar a través de **adb shell**.

Más información: [https://developer.android.com/reference/android/view/KeyEvent.html#constants\\_1](https://developer.android.com/reference/android/view/KeyEvent.html#constants_1).

- Envía un código de evento para un solo carácter a un dispositivo Android:

```
input keyevent {{codigo_evento}}
```

- Envía un texto a un dispositivo Android (%s representa espacios):

```
input text "{{texto}}"
```

- Envía una pulsación a un dispositivo Android:

```
input tap {{x_pos}} {{y_pos}}
```

- Envía un gesto de deslizamiento a un dispositivo Android:

```
input swipe {{x_start}} {{y_start}} {{x_end}} {{y_end}}  
{{duracion_en_ms}}
```

- Envía una pulsación larga a un dispositivo Android mediante un gesto de deslizamiento:

```
input swipe {{x_pos}} {{y_pos}} {{x_pos}} {{y_pos}}  
{{duracion_en_ms}}
```

# logcat

Vuelca un registro de mensajes del sistema, incluyendo seguimientos de pila cuando ocurren errores, y mensajes informativos enviados por las aplicaciones.

Más información: <https://developer.android.com/tools/logcat>.

- Muestra registros del sistema:

```
logcat
```

- Escribe registros del sistema a un archivo:

```
logcat -f {{ruta/al/archivo}}
```

- Muestra registros que coincidan con una expresión regular:

```
logcat --regex {{expresión_regular}}
```

- Muestra registros de un proceso específico:

```
logcat --pid {{pid}}
```

- Muestra registros del proceso de un paquete específico:

```
logcat --pid $(pidof -s {{paquete}})
```

# pkg

Utilidad de gestión de paquetes para Termux.

Más información: [https://wiki.termux.com/wiki/Package\\_Management](https://wiki.termux.com/wiki/Package_Management).

- Actualiza todos los paquetes instalados:

```
pkg upgrade
```

- Instala un paquete:

```
pkg install {{paquete}}
```

- Desinstala un paquete:

```
pkg uninstall {{paquete}}
```

- Reinstala un paquete:

```
pkg reinstall {{paquete}}
```

- Busca un paquete:

```
pkg search {{paquete}}
```

# pm

Muestra información sobre aplicaciones en un dispositivo Android.

Más información: <https://developer.android.com/tools/adb#pm>.

- Genera una lista de todas las aplicaciones instaladas:

```
pm list packages
```

- Genera una lista de todas las aplicaciones del sistema instaladas:

```
pm list packages -s
```

- Genera una lista de todas las aplicaciones de terceros instaladas:

```
pm list packages -3
```

- Genera una lista de aplicaciones que coinciden con determinadas palabras clave:

```
pm list packages {{palabras_clave}}
```

- Imprime la ruta del APK de una aplicación específica:

```
pm path {{app}}
```

# screencap

Toma una captura de pantalla de una pantalla móvil.

Este comando solo se puede usar a través de **adb shell**.

Más información: <https://developer.android.com/tools/adb#screencap>.

- Toma una captura de pantalla:

```
screencap {{ruta/al/archivo}}
```

# settings

Muestra información sobre el sistema operativo Android.

Más información: <https://adbinstaller.com/commands/adb-shell-settings-5b670d5ee7958178a2955536>.

- Muestra una lista de configuraciones en el espacio de nombres `global`:

```
settings list {{global}}
```

- Obtén el valor de una configuración específica:

```
settings get {{global}} {{airplane_mode_on}}
```

- Establece el valor de un ajuste:

```
settings put {{system}} {{screen_brightness}} {{42}}
```

- Elimina un ajuste específico:

```
settings delete {{secure}} {{screensaver_enabled}}
```

# wm

Muestra información sobre la pantalla de un dispositivo Android.

Este comando solo se puede usar a través de **adb shell**.

Más información: <https://adbinstaller.com/commands/adb-shell-wm-5b672b17e7958178a2955538>.

- Muestra el tamaño físico de la pantalla de un dispositivo Android:

```
wm size
```

- Muestra la densidad física de la pantalla de un dispositivo Android:

```
wm density
```



Common

# Exclamation mark

Bash incorporado para sustituir con un comando encontrado en la historia.

Más información: <https://gnu.org/software/bash/manual/bash.html#Event-Designators>.

- Sustituye con el comando anterior y lo ejecuta con sudo:

```
sudo !!
```

- Sustituye con un comando basado en un número de línea encontrado con history:

```
!{{número}}
```

- Sustituye con un comando que se utilizó un número especificado de líneas atrás:

```
!-{{número}}
```

- Sustituye por el comando más reciente que empiece por cadena:

```
!{{cadena}}
```

- Ejecuta un comando con los argumentos del último comando:

```
{{comando}} !*
```

# 2to3

Conversión automatizada de código Python 2 a 3.

Más información: <https://docs.python.org/3/library/2to3.html>.

- Muestra los cambios que se realizarían sin realizarlos (simulacro):

```
2to3 {{ruta/a/archivo.py.py}}
```

- Convierte un archivo Python 2 a Python 3:

```
2to3 --write {{ruta/a/archivo.py}}
```

- Convierte funciones específicas del lenguaje Python 2 a Python 3:

```
2to3 --write {{ruta/a/archivo.py}} --fix {{raw_input}} --fix {{print}}
```

- Convierte todas las funciones del lenguaje Python 2 excepto las especificadas a Python 3:

```
2to3 --write {{ruta/a/archivo.py}} --nofix {{has_key}} --nofix {{isinstance}}
```

- Muestra una lista de todas las características disponibles del lenguaje que se pueden convertir de Python 2 a Python 3:

```
2to3 --list-fixes
```

- Convierte todos los archivos Python 2 en un directorio a Python 3:

```
2to3 --output-dir {{ruta/a/directorio_python3}} --write-unchanged-files --nobackups {{ruta/a/directorio_python2}}
```

- Ejecuta 2to3 con varios subprocessos:

```
2to3 --processes {{1..infinity}} --output-dir {{ruta/a/directorio_python3}} --write --nobackups --no-diff {{ruta/a/directorio_python2}}
```

# 7z

Un compresor de archivos con un alto ratio de compresión.

Más información: <https://manned.org/7z>.

- [a]ñade un fichero o directorio a un archivo comprimido nuevo o existente:

```
7z a {{archivo_comprimido.7z}} {{ruta/al/  
archivo_o_directorio_a_comprimir}}
```

- Encripta un archivo comprimido existente (incluyendo los nombres de los archivos):

```
7z a {{archivo_encriptado.7z}} -p{{contraseña}} -mhe=on  
{{archivo_comprimido.7z}}
```

- E[x]trae un archivo comprimido preservando la estructura de directorios original:

```
7z x {{archivo_comprimido.7z}}
```

- E[x]trae un archivo comprimido a un directorio específico:

```
7z x {{archivo_comprimido.7z}} -o{{ruta/donde/extraer}}
```

- E[x]trae un archivo comprimido a stdout:

```
7z x {{archivo_comprimido.7z}} -so
```

- [a]rchiva usando un tipo de archivo comprimido específico:

```
7z a -t{{7z|bzip2|gzip|lzip|tar|zip}} {{archivo_comprimido}}  
{{ruta/al/archivo_o_directorio_a_comprimir}}
```

- Lista el contenido de un archivo comprimido:

```
7z l {{archivo_comprimido.7z}}
```

- Establece el nivel de compresión (entre mayor sea este, la compresión será más lenta):

```
7z a {{ruta/al/archivo_comprimido.7z}} -mx={{0|1|3|5|7|9}}  
{{ruta/al/archivo_o_directorio_a_comprimir}}
```

# 7za

Archivador de archivos con una alta relación de compresión.

Similar a **7z**, salvo que admite menos tipos de archivos pero es multiplataforma.

Más información: <https://manned.org/7za>.

- [a]rchiva un archivo o directorio:

```
7za a {{ruta/al/archivo.7z}} {{ruta/al/archivo_o_directorio}}
```

- Encripta un archivo existente (incluyendo nombres de archivos):

```
7za a {{ruta/al/encriptado.7z}} -p{{contraseña}} -mhe={{on}}  
{{ruta/al/archivo.7z}}
```

- E[x]trae un archivo preservando la estructura de directorios originales:

```
7za x {{ruta/al/archivo.7z}}
```

- E[x]trae un archivo a un directorio específico:

```
7za x {{ruta/al/archivo.7z}} -o{{ruta/de/salida}}
```

- E[x]trae un archivo a stdout:

```
7za x {{ruta/al/archivo.7z}} -so
```

- [a]rchiva usando un tipo de archivo específico:

```
7za a -t{{7z|bzip2|gzip|lzip|tar|...}} {{ruta/al/archivo.7z}}  
{{ruta/al/archivo_o_directorio}}
```

- [l]ista los contenidos de un archivo:

```
7za l {{ruta/al/archivo.7z}}
```

# 7zr

Archivador de ficheros con un alto ratio de compresión.

Similar a **7z** excepto que sólo soporta ficheros 7z.

Más información: <https://manned.org/7zr>.

- [a]rchiva un archivo o directorio:

```
7zr a {{ruta/al/archivo.7z}} {{ruta/al/archivo_o_directorio}}
```

- Cifra un archivo existente (incluidos los nombres de los archivos):

```
7zr a {{ruta/al/archivo.7z}} -p{{contraseña}} -mhe={{on}}  
{{ruta/al/archivo.7z}}
```

- E[x]trae un archivo conservando la estructura de directorios original:

```
7zr x {{ruta/al/archivo.7z}}
```

- E[x]trae un archivo a un directorio específico:

```
7zr x {{ruta/al/archivo.7z}} -o{ruta/de/salida}}
```

- E[x]trae un archivo a stdout:

```
7zr x {{ruta/al/archivo.7z}} -so
```

- [l]ista el contenido de un archivo:

```
7zr l {{ruta/al/archivo.7z}}
```

[

Comprueba los tipos de archivo y compara los valores.

Devuelve un estado de 0 si la condición se evalúa como verdadera, 1 si se evalúa como falsa.

Más información: <https://gnu.org/software/bash/manual/bash.html#index-test>.

- Comprueba si una variable dada es igual/no es igual a la cadena especificada:

```
[ "${variable}" {=|!=} "${cadena}" ]
```

- Comprueba si una variable dada es [e]qual/[n]ot [e]qual/[g]reater [t]han/[l]ess [t]han/[g]reater que o [e]qual/[l]ess que o [e]qual al número especificado:

```
[ "${variable}" -{eq|ne|gt|lt|ge|le} {{entero}} ]
```

- Comprueba si la variable especificada tiene un valor [n]o-vacío:

```
[ -n "${variable}" ]
```

- Comprueba si la variable especificada tiene un valor vacío:

```
[ -z "${variable}" ]
```

- Comprueba si el archivo [f]ile especificado existe:

```
[ -f {{ruta/al/archivo}} ]
```

- Comprueba si existe el [d]irectorio especificado:

```
[ -d {{ruta/al/directorio}} ]
```

- Comprueba si existe el archivo o directorio especificado:

```
[ -e {{ruta/al/archivo_o_directorio}} ]
```

# [[

Comprueba los tipos de archivo y compara los valores.

Devuelve 0 si la condición es verdadera, 1 si es falsa.

Más información: [https://gnu.org/software/bash/manual/bash.html#index-005b\\_005b](https://gnu.org/software/bash/manual/bash.html#index-005b_005b).

- Comprueba si una variable dada es igual/no igual a la cadena especificada:

```
[[ ${variable} {==|!=} "{{cadena}}" ]]
```

- Comprueba si una cadena dada se ajusta al glob/regex especificado:

```
[[ ${variable} {==|=~} {patron} ]]
```

- Comprueba si una variable dada es [e]qual/[n]ot [e]qual/[g]reater [t]han/[l]ess [t]han/[g]reater than o [e]qual/[l]ess than o [e]qual al número especificado:

```
[[ ${variable} -{eq|ne|gt|lt|ge|le} {integer} ]]
```

- Comprueba si la variable especificada tiene un valor [n]o-vacío:

```
[[ -n ${variable} ]]
```

- Comprueba si la variable especificada tiene un valor vacío:

```
[[ -z ${variable} ]]
```

- Comprueba si el [f]ile especificado existe:

```
[[ -f {{ruta/al/archivo}} ]]
```

- Comprueba si existe el [d]irectorio especificado:

```
[[ -d {{ruta/al/directorio}} ]]
```

- Comprueba si [e]xiste el archivo o directorio especificado:

```
[[ -e {{ruta/al/archivo_o_directorio}} ]]
```



# a2ping

Convierte imágenes en archivos EPS o PDF.

Más información: <https://manned.org/a2ping>.

- Convierte una imagen a PDF (Nota: Especificar un nombre de archivo de salida es opcional):

```
a2ping {{ruta/al/imagen.ext}} {{ruta/al/salida.pdf}}
```

- Comprime el documento utilizando el método especificado:

```
a2ping --nocompress {{none|zip|best|flate}} {{ruta/al/archivo}}
```

- Escanea HiResBoundingBox si está presente (Nota: por defecto es sí):

```
a2ping --nohires {{ruta/al/archivo}}
```

- Permite el contenido de la página por debajo y a la izquierda del origen (Nota: por defecto es no):

```
a2ping --below {{ruta/al/archivo}}
```

- Pasa argumentos adicionales a gs:

```
a2ping --gsextra {{argumentos}} {{ruta/al/archivo}}
```

- Pasa argumentos adicionales a un programa externo (por ejemplo, pdftops):

```
a2ping --extra {{argumentos}} {{ruta/al/archivo}}
```

- Muestra ayuda:

```
a2ping -h
```

# aapt

Herramienta para empaquetado de activos de Android.

Compila y empaqueta recursos de una app de Android.

Más información: <https://manned.org/aapt>.

- Lista los archivos contenidos en un archivo APK:

```
aapt list {{ruta/al/app.apk}}
```

- Muestra la metadata de una app (versión, permisos, etc.):

```
aapt dump badging {{ruta/al/app.apk}}
```

- Crea un nuevo archivo APK con archivos de un directorio especificado:

```
aapt package -F {{ruta/al/app.apk}} {{ruta/al/directorio}}
```

# ab

Herramienta comparativa del servidor Apache HTTP.

Más información: <https://httpd.apache.org/docs/current/programs/ab.html>.

- Ejecuta 100 solicitudes HTTP GET a una URL dada:

```
ab -n 100 {{url}}
```

- Ejecuta 100 solicitudes HTTP GET, en lotes simultáneos de a 10, a una URL:

```
ab -n 100 -c 10 {{url}}
```

- Ejecuta 100 solicitudes HTTP POST a una URL, utilizando la carga JSON de un archivo:

```
ab -n 100 -T {{application/json}} -p {{ruta/al/archivo.json}}  
{{url}}
```

- Utiliza HTTP [K]eep Alive, es decir, realiza múltiples solicitudes dentro de una sesión HTTP:

```
ab -k {{url}}
```

- Establece el máximo número de segundos utilizados para la comparación:

```
ab -t {{60}} {{url}}
```

# abduco

Administrador de sesión de terminal.

Más información: <https://www.brain-dump.org/projects/abduco/>.

- Lista sesiones:

```
abduco
```

- Adjunta a una sesión, creándola si no existe:

```
abduco -A {{nombre}} {{bash}}
```

- Adjunta a una sesión con `dvtm`, creándola si no existe:

```
abduco -A {{nombre}}
```

- Separarse de una sesión:

```
<Ctrl> + \
```

- Adjunta a una sesión en modo solo-lectura:

```
abduco -Ar {{nombre}}
```

# ac

Imprime estadísticas sobre cuanto tiempo han estado conectados los usuarios.

Más información: <https://man.openbsd.org/ac>.

- Imprime cuanto tiempo ha estado conectado el usuario actual en horas:

```
ac
```

- Imprime cuanto tiempo han estado conectados los usuarios en horas:

```
ac -p
```

- Imprime cuanto tiempo ha estado conectado un usuario en particular en horas:

```
ac -p {{nombre_de_usuario}}
```

- Imprime cuánto tiempo ha estado conectado un usuario en particular en horas por día (con total):

```
ac -dp {{nombre_de_usuario}}
```

# accelerate

Una biblioteca que permite ejecutar el mismo código PyTorch en cualquier configuración distribuida.

Más información: <https://huggingface.co/docs/accelerate/index>.

- Imprime información del entorno:

```
accelerate env
```

- Crea interactivamente un archivo de configuración:

```
accelerate config
```

- Imprime el coste estimado en memoria de la GPU al ejecutar un modelo Hugging Face con distintos tipos de datos:

```
accelerate estimate-memory {{nombre/modelo}}
```

- Prueba un archivo de configuración de Accelerate:

```
accelerate test --config_file {{ruta/a/config.yaml}}
```

- Ejecuta un modelo en CPU con Accelerate:

```
accelerate launch {{ruta/a/script.py}} {{--cpu}}
```

- Ejecuta un modelo en multi-GPU con Accelerate, con 2 máquinas:

```
accelerate launch {{ruta/a/script.py}} --multi_gpu --  
num_machines 2
```

# ack

Una herramienta de búsqueda como grep, optimizada para desarrolladores.

Vea también: **rg**, que es más rápido.

Más información: <https://beyondgrep.com/documentation>.

- Busca archivos que contengan una cadena o expresión regular en el directorio actual de forma recursiva:

```
ack "{{patrón_de_búsqueda}}"
```

- Busca un patrón sin distinción entre mayúsculas y minúsculas:

```
ack --ignore-case "{{patrón_de_búsqueda}}"
```

- Busca líneas que coincidan con un patrón, imprimiendo s[o]lamente el texto coincidente y no el resto de la línea:

```
ack -o "{{patrón_de_búsqueda}}"
```

- Limita la búsqueda a archivos de un tipo específico:

```
ack --type {{ruby}} "{{patrón_de_búsqueda}}"
```

- Busca archivos que no sean de un cierto tipo:

```
ack --type no{{ruby}} "{{patrón_de_búsqueda}}"
```

- Cuenta el número total de coincidencias encontradas:

```
ack --count --no-filename "{{patrón_de_búsqueda}}"
```

- Imprime sólo los nombres de los archivos y el número de coincidencias de cada archivo:

```
ack --count --files-with-matches "{{patrón_de_búsqueda}}"
```

- Lista todos los valores que se pueden utilizar con `--type`:

```
ack --help-types
```

# acme.sh --dns

Utiliza un desafío DNS-01 para emitir un certificado TLS.

Más información: <https://github.com/acmesh-official/acme.sh/wiki>.

- Emite un certificado utilizando un modo API DNS automático:

```
acme.sh --issue --dns {{gnd_gd}} --domain {{ejemplo.com}}
```

- Emite un certificado comodín (marcado con un asterisco) utilizando un modo API DNS automático:

```
acme.sh --issue --dns {{dns_namesilo}} --domain  
{{ejemplo.com}} --domain {{*.ejemplo.com}}
```

- Emite un certificado utilizando un modo de alias DNS:

```
acme.sh --issue --dns {{dns_cf}} --domain {{ejemplo.com}} --  
challenge-alias {{alias-para-ejemplo-validacion.com}}
```

- Emite un certificado mientras se desactiva el sondeo automático de Cloudflare / Google DNS después de añadir el registro DNS especificando un tiempo de espera personalizado en segundos:

```
acme.sh --issue --dns {{dns_namecheap}} --domain  
{{ejemplo.com}} --dnssleep {{300}}
```

- Emite un certificado utilizando un modo DNS manual:

```
acme.sh --issue --dns --domain {{ejemplo.com}} --yes-I-know-  
dns-manual-mode-enough-go-ahead-please
```



# acme.sh

Shell script implementando el protocolo cliente ACME, una alternativa a **certbot**.

Vea también **acme.sh dns**.

Más información: <https://github.com/acmesh-official/acme.sh>.

- Emite un certificado usando el modo webroot:

```
acme.sh --issue --domain {{ejemplo.com}} --webroot {{ruta/al/webroot}}
```

- Emite un certificado para múltiples dominios utilizando el modo autónomo a través del puerto 80:

```
acme.sh --issue --standalone --domain {{ejemplo.com}} --domain {{www.ejemplo.com}}
```

- Emite un certificado en modo autónomo TLS utilizando el puerto 443:

```
acme.sh --issue --alpn --domain {{ejemplo.com}}
```

- Emite un certificado utilizando una configuración de Nginx operativa:

```
acme.sh --issue --nginx --domain {{ejemplo.com}}
```

- Emite un certificado utilizando una configuración de Apache operativa:

```
acme.sh --issue --apache --domain {{ejemplo.com}}
```

- Emite un certificado comodín (\*) utilizando un modo API DNS automático:

```
acme.sh --issue --dns {{dns_cf}} --domain {{*.ejemplo.com}}
```

- Instala archivos de certificado en las ubicaciones especificadas (útil para la renovación automática de certificados):

```
acme.sh --install-cert -d {{ejemplo.com}} --key-file {{ruta/al/ejemplo.com.key}} --fullchain-file {{ruta/al/ejemplo.com.cer}} --reloadcmd {"systemctl force-reload nginx"}
```

# act

Ejecuta acciones de GitHub localmente mediante Docker.

Más información: <https://github.com/nektos/act>.

- Lista las acciones disponibles:

```
act -l
```

- Ejecuta el evento por defecto:

```
act
```

- Ejecuta un evento específico:

```
act {{event_type}}
```

- Ejecuta una acción específica:

```
act -a {{identificador_de_acción}}
```

- Simula una acción:

```
act -n
```

- Muestra registros detallados:

```
act -v
```

# acyclic

Haz un gráfico acíclico invirtiendo algunos bordes.

Filtros Graphviz: **acyclic**, **bcomps**, **comps**, **edgepaint**, **gvcolor**, **gvpack**, **mingle**, **nop**, **sccmap**, **tred**, & **unflatten**.

Más información: <https://graphviz.org/pdf/acyclic.1.pdf>.

- Haz un gráfico acíclico invirtiendo algunos bordes:

```
acyclic {{ruta/a/entrada.gv}} > {{ruta/a/salida.gv}}
```

- Imprime si un gráfico es acíclico, tiene un ciclo o si no posee instrucciones, no genera ningún gráfico de salida:

```
acyclic -v -n {{ruta/a/entrada.gv}}
```

- Muestra información de ayuda:

```
acyclic -?
```

# adb install

Instalación de Android Debug Bridge: Envía paquetes a una instancia del emulador de Android o a dispositivos Android conectados.

Más información: <https://developer.android.com/tools/adb>.

- Envía una aplicación Android a un emulador/dispositivo:

```
adb install {{ruta/al/archivo.apk}}
```

- Envía una aplicación Android a un emulador/dispositivo específico (ignora \$ANDROID\_SERIAL):

```
adb -s {{numero_de_serie}} install {{ruta/al/archivo.apk}}
```

- Reinstala una aplicación existente, manteniendo sus datos:

```
adb install -r {{ruta/al/archivo.apk}}
```

- Envía una aplicación Android permitiendo bajar el código de versión (sólo paquetes depurables):

```
adb install -d {{ruta/al/archivo.apk}}
```

- Concede todos los permisos enumerados en el manifiesto de la aplicación:

```
adb install -g {{ruta/al/archivo.apk}}
```

- Actualiza rápidamente un paquete instalado actualizando sólo las partes del APK que han cambiado:

```
adb install --fastdeploy {{ruta/al/archivo.apk}}
```

# adb logcat

Vuelca un registro de mensajes del sistema.

Más información: <https://developer.android.com/tools/logcat>.

- Muestra registros del sistema:

```
adb logcat
```

- Muestra las líneas que coincidan con una expresión regular:

```
adb logcat -e {{expresión_regular}}
```

- Muestra los registros de una etiqueta en un modo específico ([V]erboso, [D]epuración, [I]nformación, [W]arning, [E]rror, [F]atal, [S]ilencioso), filtrando otras etiquetas:

```
adb logcat {{etiqueta}}:{{modo}} *:S
```

- Muestra los registros de aplicaciones React Native en modo [V]erboso [S]ilenciando otras etiquetas:

```
adb logcat ReactNative:V ReactNativeJS:V *:S
```

- Muestra los registros de todas las etiquetas con nivel de prioridad [W]arning y superior:

```
adb logcat *:W
```

- Muestra los registros de un proceso específico:

```
adb logcat --pid {{pid}}
```

- Muestra los registros del proceso de un paquete específico:

```
adb logcat --pid $(adb shell pidof -s {{paquete}})
```

- Colorea el registro (normalmente se usan filtros):

```
adb logcat -v color
```

# adb reverse

Android Debug Bridge Reverse: conexiones de socket inversas desde una instancia de emulador de Android o dispositivos Android conectados.

Más información: <https://developer.android.com/tools/adb>.

- Lista todas las conexiones de socket inverso de emuladores y dispositivos:

```
adb reverse --list
```

- Invierte un puerto TCP desde un emulador o dispositivo a localhost:

```
adb reverse tcp:{{remote_port}} tcp:{{local_port}}
```

- Elimina una conexión de socket inversa de un emulador o dispositivo:

```
adb reverse --remove tcp:{{remote_port}}
```

- Elimina todas las conexiones de socket inverso de todos los emuladores y dispositivos:

```
adb reverse --remove-all
```

# adb shell

Android Debug Bridge Shell: Ejecuta comandos shell remotos en una instancia del emulador de Android o en dispositivos Android conectados.

Más información: <https://developer.android.com/tools/adb>.

- Inicia una shell interactiva remota en el emulador o dispositivo:

```
adb shell
```

- Obtén todas las propiedades del emulador o dispositivo:

```
adb shell getprop
```

- Revierte todos los permisos de ejecución a sus valores por defecto:

```
adb shell pm reset-permissions
```

- Revoca un permiso peligroso para una aplicación:

```
adb shell pm revoke {{paquete}} {{permission}}
```

- Activa un evento de clave:

```
adb shell input keyevent {{keycode}}
```

- Borra los datos de una aplicación en un emulador o dispositivo:

```
adb shell pm clear {{paquete}}
```

- Inicia una actividad en el emulador o dispositivo:

```
adb shell am start -n {{paquete}}/{{activity}}
```

- Inicia la actividad de inicio en un emulador o dispositivo:

```
adb shell am start -W -c android.intent.category.HOME -a  
android.intent.action.MAIN
```

# adb

Android Debug Bridge: comunica con una instancia de un emulador Android o conecta dispositivos Android.

Algunos subcomandos, como **shell**, tienen su propia documentación de uso.

Más información: <https://developer.android.com/tools/adb>.

- Verifica si el proceso del servidor adb está ejecutandose y lo inicia:

```
adb start-server
```

- Termina el proceso del servidor adb:

```
adb kill-server
```

- Inicia una terminal remota en la instancia del emulador/dispositivo de destino:

```
adb shell
```

- Instala una aplicación Android a un emulador/dispositivo:

```
adb install -r {{ruta/al/archivo.apk}}
```

- Copia un archivo/directorio desde el dispositivo de destino:

```
adb pull {{ruta/al/archivo_o_directorio_en_el_dispositivo}}  
{{ruta/al/directorio_de_destino_local}}
```

- Copia un archivo/directorio al dispositivo de destino:

```
adb push {{ruta/al/archivo_o_directorio_local}} {{ruta/al/  
directorio_de_destino_en_el_dispositivo}}
```

- Obtén una lista de dispositivos conectados:

```
adb devices
```



# adscript

Compilador de archivos Adscript.

Más información: <https://github.com/Amplus2/Adscript>.

- Compila un archivo en un archivo objeto:

```
adscript --output {{ruta/al/archivo.o}} {{ruta/al/  
archivo_de_entrada.adscript}}
```

- Compila y vincula un archivo a un ejecutable independiente:

```
adscript --executable --output {{ruta/a/archivo}} {{ruta/a/  
archivo_entrada.adscript}}
```

- Compila un archivo a LLVM IR en lugar de código de máquina nativo:

```
adscript --llvm-ir --output {{ruta/a/archivo.ll}} {{ruta/a/  
archivo_entrada.adscript}}
```

- Crea un archivo con código objeto para otra arquitectura u otro sistema operativo:

```
adscript --target-triple {{i386-linux-elf}} --output {{ruta/  
a/archivo.o}} {{ruta/a/archivo_entrada.adscript}}
```

# afconvert

Convierte entre formatos de archivo AFF y raw.

Más información: <https://manned.org/afconvert.1>.

- Utiliza una extensión específica (predeterminado: aff):

```
afconvert -a {{extension}} {{ruta/al/archivo_de_entrada}}  
{{ruta/al/archivo_salida1 ruta/al/archivo_salida2 ...}}
```

- Utiliza un nivel de compresión específico (predeterminado: 7):

```
afconvert -X{{0..7}} {{ruta/al/archivo_de_entrada}} {{ruta/  
al/archivo_salida1 ruta/al/archivo_salida2 ...}}
```

# ag

The Silver Searcher. Como **ack**, pero apunta a ser más rápido.

Más información: [https://github.com/ggreer/the\\_silver\\_searcher](https://github.com/ggreer/the_silver_searcher).

- Encuentra archivos que contengan "foo", e imprime las líneas coincidentes en su contexto:

```
ag {{foo}}
```

- Busca archivos que contengan "foo" en un directorio específico:

```
ag {{foo}} {{ruta/al/directorio}}
```

- Busca archivos que contengan "foo", pero sólo se alistan los nombres de los archivos:

```
ag -l {{foo}}
```

- Busca archivos que contengan "FOO" sin distinguir entre mayúsculas y minúsculas, e imprime sólo la coincidencia, en lugar de toda la línea:

```
ag -i -o {{F00}}
```

- Busca "foo" en archivos cuyo nombre coincide con "bar":

```
ag {{foo}} -G {{bar}}
```

- Busca archivos cuyo contenido coincide con una expresión regular:

```
ag '{{^ba(r|z)$}}'
```

- Busca archivos con un nombre que coincida con "foo":

```
ag -g {{foo}}
```

# agate

Un sencillo servidor para el protocolo de red Gemini.

Más información: <https://github.com/mbrubeck/agate>.

- Ejecuta y genera una clave privada y un certificado:

```
agate --content {{ruta/a/contenido/}} --addr {[::]:1965} --  
addr {{0.0.0.0:1965}} --hostname {{ejemplo.com}} --lang {{sp-  
SP}}
```

- Ejecuta servidor:

```
agate {{ruta/al/archivo}}
```

- Muestra la ayuda:

```
agate -h
```

# age

Una herramienta de encriptación de archivos sencilla, moderna y segura.

Más información: <https://github.com/FiloSottile/age>.

- Genera un archivo cifrado que se puede descifrar con una frase de contraseña:

```
age --passphrase --output {{ruta/al/archivo_encriptado}}  
{{ruta/al/archivo_no_encriptado}}
```

- Genera un par de claves, guardando la clave privada en un archivo no cifrado e imprimiendo la clave pública en stdout:

```
age-keygen --output {{ruta/al/archivo}}
```

- Cifra un archivo con una o más claves públicas que se introducen como literales:

```
age --recipient {{clave_publica_1}} --recipient  
{{clave_publica_2}} {{ruta/al/archivo_sin_cifrar}} --output  
{{ruta/al/archivo_cifrado}}
```

- Cifra un archivo con una o varias claves públicas especificadas en un archivo de destinatarios:

```
age --recipients-file {{ruta/al/archivo_recipientes}} {{ruta/  
para/archivo_sin_cifrar}} --output {{ruta/al/  
archivo_encriptado}}
```

- Descifra un archivo con una frase de contraseña:

```
age --decrypt --output {{ruta/al/archivo_descifrado}} {{ruta/  
para/archivo_cifrado}}
```

- Descifra un archivo con un archivo de clave privada:

```
age --decrypt --identity {{ruta/al/archivo_de_clave_privada}}  
--output {{ruta/para/archivo_descifrado}} {{ruta/para/  
archivo_cifrado}}
```

# aircrack-ng

Crackea claves WEP y WPA/WPA2 desde handshake en paquetes capturados.

Parte de la suite de software de red Aircrack-ng.

Más información: <https://www.aircrack-ng.org/doku.php?id=aircrack-ng>.

- Crackea la clave desde el archivo de captura usando [w]ordlist:

```
aircrack-ng -w {{ruta/al/lista.txt}} {{ruta/al/captura.cap}}
```

- Descifra la clave del archivo de captura utilizando [w]ordlist y el [e]ssid del punto de acceso:

```
aircrack-ng -w {{ruta/al/lista.txt}} -e {{ssid}} {{ruta/al/captura.cap}}
```

- Descifra la clave del archivo de captura utilizando [w]ordlist y la dirección MAC del punto de acceso:

```
aircrack-ng -w {{ruta/al/lista.txt}} --bssid {{mac}} {{ruta/al/captura.cap}}
```

# aireplay-ng

Inyecta paquetes en una red inalámbrica.

Parte de **aircrack-ng**.

Más información: <https://www.aircrack-ng.org/doku.php?id=aireplay-ng>.

- Envía una cantidad específica de paquetes disociados dada la dirección MAC de un punto de acceso, la dirección MAC de un cliente y una interfaz:

```
sudo aireplay-ng --deauth {{cantidad}} --bssid  
{{mac_punto_acceso}} --dmac {{mac_cliente}} {{interfaz}}
```

# airmon-ng

Activa el modo monitor en dispositivos de red inalámbricos.

Parte de **aircrack-ng**.

Más información: <https://www.aircrack-ng.org/doku.php?id=airmon-ng>.

- Lista dispositivos inalámbricos y sus estados:

```
sudo airmon-ng
```

- Activa el modo monitor para un dispositivo específico:

```
sudo airmon-ng start {{wlan0}}
```

- Elimina los procesos perturbadores que utilizan dispositivos inalámbricos:

```
sudo airmon-ng check kill
```

- Desactiva el modo monitor para una interfaz de red específica:

```
sudo airmon-ng stop {{wlan0mon}}
```



# airodump-ng

Captura paquetes y muestra información sobre redes inalámbricas.

Parte de **aircrack-ng**.

Más información: <https://www.aircrack-ng.org/doku.php?id=airodump-ng>.

- Captura paquetes y muestra información sobre una red inalámbrica:

```
sudo airodump-ng {{interface}}
```

- Captura paquetes y muestra información sobre una red inalámbrica dada la dirección MAC y canal, y guarda la salida en un archivo:

```
sudo airodump-ng --channel {{canal}} --write {{ruta/al/archivo}} --bssid {{mac}} {{interfaz}}
```

# airpaste

Comparte mensajes y archivos sobre la misma red usando mDNS.

Más información: <https://github.com/mafintosh/airpaste>.

- Espera un mensaje y lo muestra cuando se reciba:

```
airpaste
```

- Envía un texto:

```
echo {{texto}} | airpaste
```

- Envía un archivo:

```
airpaste < {{ruta/al/archivo}}
```

- Recibe un archivo:

```
airpaste > {{ruta/al/archivo}}
```

- Crea un canal o se une al mismo:

```
airpaste {{nombre_canal}}
```

# ajson

Ejecuta JSONPath en objetos JSON.

Más información: <https://github.com/spyzhov/ajson>.

- Lee JSON de un archivo y ejecuta una expresión JSONPath especificada:

```
ajson '{{$.json[?(@.path)]}}' {{ruta/al/archivo.json}}
```

- Lee JSON de `stdin` y ejecuta una expresión JSONPath especificada:

```
cat {{ruta/al/archivo.json}} | ajson '{{$.json[?(@.path)]}}'
```

- Lee JSON de una URL y evalúa una expresión JSONPath especificada:

```
ajson '{{avg($.price)}}' '{{https://ejemplo.com/api}}'
```

- Lee un simple cadena JSON y calcula un valor:

```
echo '{{3}}' | ajson '{{2 * pi * $}}'
```

# alacritty

Emulador de terminal acelerado por GPU y multiplataforma.

Más información: <https://github.com/alacritty/alacritty>.

- Abre una nueva ventana de Alacritty:

```
alacritty
```

- Ejecuta Alacritty en un directorio específico:

```
alacritty --working-directory {{ruta/al/directorio}}
```

- Ejecuta un comando en una nueva ventana de Alacritty:

```
alacritty -e {{comando}}
```

- Especifica un archivo de configuración alternativo (por defecto es `$XDG_CONFIG_HOME/alacritty/alacritty.toml`):

```
alacritty --config-file {{ruta/al/config.toml}}
```

- Ejecuta con recarga automática de la configuración activada (puede activarse por defecto en `alacritty.toml`):

```
alacritty --live-config-reload --config-file {{ruta/al/config.toml}}
```

# alex

Una herramienta que detecta escritura insensible y desconsiderada.

Ayuda a encontrar en el texto frases que son parciales con el género, que polarizan, o están relacionadas con la raza, son desconsideradas con la religión u otras frases tendenciosas.

Más información: <https://github.com/get-alex/alex>.

- Analiza texto desde `stdin`:

```
echo {{His network looks good}} | alex --stdin
```

- Analiza todos los archivos del directorio actual:

```
alex
```

- Analiza un archivo dado:

```
alex {{ruta/al/archivo_de_texto.md}}
```

- Analiza todos los archivos Markdown excepto `ejemplo.md`:

```
alex *.md !{{ruta/hacia/ejemplo.md}}
```

# alias

Crea alias -- palabras que son reemplazadas por una cadena de comando(s).

Los alias son temporales en la sesión de shell actual, a no ser que estén definidos en el archivo de configuración de la shell, ej. `~/ .bashrc`.

Más información: <https://tldp.org/LDP/abs/html/aliases.html>.

- Lista todos los alias:

```
alias
```

- Crea un alias genérico:

```
alias {{nombre}}="{{comando}}"
```

- Muestra el comando asociado a un alias:

```
alias {{nombre}}
```

- Elimina un alias (con el comando asociado):

```
unalias {{nombre}}
```

- Convierte `rm` en un comando interactivo:

```
alias {{rm}}="{{rm -i}}"
```

- Define `la` como un atajo para `ls -a`:

```
alias {{la}}="{{ls -a}}"
```

# amass enum

Busca subdominios de un dominio.

Más información: [https://github.com/owasp-amass/amass/blob/master/doc/user\\_guide.md#the-enum-subcommand](https://github.com/owasp-amass/amass/blob/master/doc/user_guide.md#the-enum-subcommand).

- Búsqueda pasiva de subdominios de un dominio:

```
amass enum -passive -d {{nombre_de_dominio}}
```

- Busca subdominios de un dominio y los verifica activamente intentando resolver los subdominios encontrados:

```
amass enum -active -d {{nombre_de_dominio}} -p  
{{80,443,8080}}
```

- Hace una búsqueda en su modalidad fuerza bruta de subdominios:

```
amass enum -brute -d {{nombre_de_dominio}}
```

- Guarda los resultados en un archivo de texto:

```
amass enum -o {{archivo_salida}} -d {{nombre_de_dominio}}
```

- Guarda los resultados a una base de datos:

```
amass enum -o {{archivo_salida}} -dir {{ruta/a/  
directorio_base_de_datos}}
```

# amass intel

Recopila información de código abierto sobre una organización, como dominios raíz y ASNs.

Más información: [https://github.com/owasp-amass/amass/blob/master/doc/user\\_guide.md#the-intel-subcommand](https://github.com/owasp-amass/amass/blob/master/doc/user_guide.md#the-intel-subcommand).

- Encuentra dominios raíz en un rango de direcciones IP específico:

```
amass intel -addr {{192.168.0.1-254}}
```

- Usa métodos activos de reconocimiento:

```
amass intel -active -addr {{192.168.0.1-254}}
```

- Encuentra dominios raíz relacionados con un dominio específico:

```
amass intel -whois -d {{nombre_de_dominio}}
```

- Encuentra ASN pertenecientes a una organización específica:

```
amass intel -org {{nombre_de_organizacion}}
```

- Encuentra dominios raíz pertenecientes a un Número de Sistema Autónomo específico:

```
amass intel -asn {{string}}
```

- Guarda los resultados en un archivo de texto específico:

```
amass intel -o {{ruta/al/archivo_de_salida}} -whois -d {{nombre_de_dominio}}
```



# amass

Herramienta de mapeo de superficie de ataque en profundidad y descubrimiento de activos.

Algunos subcomandos como **intel** tienen su propia documentación de uso.

Más información: <https://github.com/owasp-amass/amass>.

- Ejecuta un subcomando Amass:

```
amass {{intel|enum}} {{options}}
```

- Muestra ayuda:

```
amass -help
```

- Muestra ayuda sobre un subcomando de Amass:

```
amass {{intel|enum}} -help
```

- Muestra la versión:

```
amass -version
```

# ani-cli

Un cli para navegar y ver anime.

Más información: <https://github.com/pystardust/ani-cli>.

- Busca anime por nombre:

```
ani-cli "{{nombre_del_anime}}"
```

- Descarga episodio:

```
ani-cli -d "{{nombre_del_anime}}"
```

- Usa VLC como reproductor multimedia:

```
ani-cli -v "{{nombre_del_anime}}"
```

- Especifica el episodio que desea ver:

```
ani-cli -e {{numero_episodio}} "{{nombre_del_anime}}"
```

- Continúa viendo el anime desde el historial:

```
ani-cli -c
```

- Actualiza ani-cli:

```
ani-cli -U
```

# anki

Potente e inteligente programa de flashcards.

Más información: <https://docs.ankiweb.net>.

- Inicia `anki`:

```
anki
```

- Inicia `anki` con un perfil específico:

```
anki -p {{nombre_perfil}}
```

- Inicia `anki` en un idioma específico:

```
anki -l {{idioma}}
```

- Inicia `anki` desde un directorio específico en lugar del predeterminado (`~/Anki`):

```
anki -b {{ruta/al/directorio}}
```

# ansible-doc

Muestra información sobre los módulos instalados en las bibliotecas de Ansible.

Muestra una concisa lista de complementos y sus breves descripciones.

Más información: <https://docs.ansible.com/ansible/latest/cli/ansible-doc.html>.

- Lista de complementos disponibles acorde a su acción (módulos):

```
ansible-doc --list
```

- Lista de complementos disponibles dado un tipo específico:

```
ansible-doc --type {{become|cache|callback|cliconf|  
connection|...}} --list
```

- Muestra información sobre un complemento acorde a su acción específica (módulo):

```
ansible-doc {{nombre_complemento}}
```

- Muestra información acerca de un complemento dado un tipo específico:

```
ansible-doc --type {{become|cache|callback|cliconf|  
connection|...}} {{nombre_complemento}}
```

- Muestra fragmentos de las acciones respecto al tipo de complemento y su especificidad de tipo de acción (módulos):

```
ansible-doc --snippet {{nombre_complemento}}
```

- Muestra información de acuerdo al complemento dada su especificidad de acción (módulo) como JSON:

```
ansible-doc --json {{nombre_complemento}}
```

# ansible-inventory

Muestra o vuelca un inventario de Ansible.

Vea también: **ansible**.

Más información: <https://docs.ansible.com/ansible/latest/cli/ansible-inventory.html>.

- Muestra el inventario por defecto:

```
ansible-inventory --list
```

- Muestra un inventario personalizado:

```
ansible-inventory --list --inventory {{ruta/al/  
archivo_o_script_o_directorio}}
```

- Muestra el inventario por defecto en YAML:

```
ansible-inventory --list --yaml
```

- Vuelca el inventario por defecto a un fichero:

```
ansible-inventory --list --output {{ruta/al/archivo}}
```

# ansible-pull

Extrae playbooks ansible de un repositorio VCS y los ejecuta para el host local.

Más información: <https://docs.ansible.com/ansible/latest/cli/ansible-pull.html>.

- Extrae un playbook de un VCS y ejecuta local.yml del playbook por defecto:

```
ansible-pull -U {{url_repositorio}}
```

- Extrae un playbook de un VCS y ejecuta un playbook específico:

```
ansible-pull -U {{url_repositorio}} {{playbook}}
```

- Extrae un playbook de un VCS en una rama determinada y ejecuta un playbook específico:

```
ansible-pull -U {{url_repositorio}} -C {{rama}} {{playbook}}
```

- Extrae un playbook de un VCS, en tanto especificando un archivo hosts y ejecuta un playbook específico:

```
ansible-pull -U {{url_repositorio}} -i {{archivo_hosts}}  
{{playbook}}
```

# ansible

Gestiona grupos de ordenadores de forma remota a través de SSH. (usa el fichero `/etc/ansible/hosts` para añadir nuevos grupos de hosts).

Algunos subcomandos como **galaxy** tienen su propia documentación.

Más información: <https://www.ansible.com/>.

- Lista los hosts pertenecientes a un grupo:

```
ansible {{grupo}} --list-hosts
```

- Hace ping a un grupo de hosts invocando al módulo ping:

```
ansible {{grupo}} -m ping
```

- Muestra información sobre un grupo de hosts invocando al módulo setup:

```
ansible {{grupo}} -m setup
```

- Ejecuta un comando en un grupo de hosts invocando el módulo de command con argumentos:

```
ansible {{grupo}} -m command -a '{{mi_comando}}'
```

- Ejecuta un comando con privilegios administrativos:

```
ansible {{grupo}} --become --ask-become-pass -m command -a  
'{{mi_comando}}'
```

- Ejecuta un comando utilizando un archivo de inventario personalizado:

```
ansible {{grupo}} -i {{archivo_de_inventario}} -m command -a  
'{{mi_comando}}'
```

- Lista los grupos de un inventario:

```
ansible localhost -m debug -a '{{var=groups.keys()}}'
```

# ansiweather

Un script de shell para mostrar las condiciones meteorológicas actuales en tu terminal.

Más información: <https://github.com/fcambus/ansiweather>.

- Muestra una previsión usando unidades métricas de los siguientes siete días de una ubicación:

```
ansiweather -u metric -f 7 -l {{Rzeszow,PL}}
```

- Muestra una previsión de los siguientes cinco días con símbolos e información de luz diurna de tu ubicación actual:

```
ansiweather -F -s true -d true
```

- Muestra una previsión con los datos de viento y humedad de tu ubicación actual:

```
ansiweather -w true -h true
```



# antibody

"El más rápido" administrador de complementos de shell.

Más información: <https://getantibody.github.io>.

- Empaqueta todos los complementos para su carga estática:

```
antibody bundle < {{ ~/.zsh_plugins.txt }} >  
{{ ~/.zsh_plugins.sh }}
```

- Actualiza todos los empaquetados:

```
antibody update
```

- Lista todos los complementos instalados:

```
antibody list
```

# apg

Crea contraseñas aleatorias arbitrariamente complejas.

Más información: <https://manned.org/apg>.

- Crea contraseñas aleatorias (la longitud predeterminada de la contraseña es 8):

```
apg
```

- Crea una contraseña con al menos 1 símbolo (S), 1 número (N), 1 mayúscula (C), 1 minúscula (L):

```
apg -M SNCL
```

- Crea una contraseña con 16 caracteres:

```
apg -m {{16}}
```

- Crea una contraseña con una longitud máxima de 16:

```
apg -x {{16}}
```

- Crea una contraseña que no aparece en un diccionario (se debe proporcionar el archivo del diccionario):

```
apg -r {{ruta/al/archivo_diccionario}}
```

# apkleaks

Expone URIs, endpoints y secretos de archivos APK.

Nota: APKLeaks utiliza el desensamblador **jadx** para decompilar archivos APK.

Más información: <https://github.com/dwisiswant0/apkleaks>.

- Escanea un archivo APK en busca de URIs, endpoints y secretos:

```
apkleaks --file {{ruta/al/archivo.apk}}
```

- Escanea y guarda el resultad[o] en un archivo específico:

```
apkleaks --file {{ruta/al/archivo.apk}} --output {{ruta/al/archivo.txt}}
```

- Pasar [a]rgumentos del desensamblador jadx:

```
apkleaks --file {{ruta/al/archivo.apk}} --args "{{--threads-count 5 --deobf}}"
```

# apktool

Ingeniería inversa de archivos APK.

Más información: <https://ibotpeaches.github.io/Apktool/>.

- Decodifica un archivo APK:

```
apktool d {{archivo.apk}}
```

- Construye un archivo APK desde un directorio:

```
apktool b {{ruta/al/directorio}}
```

- Instala y almacena un framework:

```
apktool if {{framework.apk}}
```

# apm

Editor Atom Package Manager.

Vea **atom**.

Más información: <https://github.com/atom/apm>.

- Instala un paquete de <http://atom.io/packages> o un tema de <http://atom.io/themes>:

```
apm install {{nombre_de_paquete}}
```

- Elimina un paquete/tema:

```
apm remove {{nombre_de_paquete}}
```

- Actualiza un paquete/tema:

```
apm upgrade {{nombre_de_paquete}}
```

# apropos

Busca nombres y descripciones en las páginas del manual.

Más información: <https://manned.org/apropos>.

- Busca una palabra clave utilizando una expresión regular:

```
apropos {{expresion_regular}}
```

- Busca sin restringir la salida al ancho de la terminal:

```
apropos -l {{expresion_regular}}
```

- Busca páginas que contengan todas las expresiones dadas:

```
apropos {{expresion_regular_1}} -a {{expresion_regular_2}} -a  
{{expresion_regular_3}}
```

# arch

Muestra el nombre de la arquitectura del sistema.

Vea también **uname**.

Más información: <https://www.gnu.org/software/coreutils/arch>.

- Muestra la arquitectura del sistema:

`arch`

# archwiki-rs

Lee, busca y descarga páginas de la ArchWiki.

Más información: <https://gitlab.com/lucifayr/archwiki-rs>.

- Lee una página de la ArchWiki:

```
archwiki-rs read-page {{título_de_página}}
```

- Lee una página de la ArchWiki en el formato especificado:

```
archwiki-rs read-page {{título_de_página}} --format {{plain-text|markdown|html}}
```

- Busca páginas en ArchWiki con el texto proporcionado:

```
archwiki-rs search "{{texto_a_buscar}}" --text-search
```

- Descarga una copia local de todas las páginas de ArchWiki en un directorio específico:

```
archwiki-rs local-wiki {{ruta/a/wiki_local}} --format {{plain-text|markdown|html}}
```



# argocd app

Interfaz de línea de comandos para gestionar aplicaciones por CD Argo.

Más información: [https://argo-cd.readthedocs.io/en/stable/user-guide/commands/argocd\\_app/](https://argo-cd.readthedocs.io/en/stable/user-guide/commands/argocd_app/).

- Lista aplicaciones:

```
argocd app list --output {{json|yaml|wide}}
```

- Obtén los detalles de la aplicación:

```
argocd app get {{nombre_de_la_aplicacion}} --output {{json|yaml|wide}}
```

- Despliega la aplicación internamente (en el mismo clúster en el que se ejecuta Argo CD):

```
argocd app create {{nombre_de_la_aplicación}} --repo {{git_repo_url}} --path {{ruta/al/repo}} --dest-server https://kubernetes.default.svc --dest-namespace {{ns}}
```

- Elimina una aplicación:

```
argocd app delete {{nombre_de_la_aplicación}}
```

- Activa la sincronización automática de aplicaciones:

```
argocd app set {{nombre_de_la_aplicacion}} --sync-policy auto --auto-prune --self-heal
```

- Previsualiza la sincronización de aplicaciones sin afectar al clúster:

```
argocd app sync {{nombre_de_la_aplicacion}} --dry-run --prune
```

- Muestra el historial de despliegue de aplicaciones:

```
argocd app history {{nombre_de_la_aplicacion}} --output {{wide|id}}
```

- Retrocede la aplicación a una versión anterior desplegada por ID de historial (eliminando recursos inesperados):

```
argocd app rollback {{nombre_de_la_aplicacion}} {{identificador_de_historial}} --prune
```

# argon2

Calcula hashes criptográficos Argon2.

Más información: <https://github.com/P-H-C/phc-winner-argon2#command-line-utility>.

- Calcula un hash con una contraseña y un salt con los parámetros por defecto:

```
echo "{{contraseña}}" | argon2 "{{texto_sal}}"
```

- Calcula un hash con el algoritmo especificado:

```
echo "{{contraseña}}" | argon2 "{{texto_sal}}" -{{d|i|id}}
```

- Muestra el hash de salida sin información adicional:

```
echo "{{contraseña}}" | argon2 "{{texto_sal}}" -e
```

- Calcula un hash con una cantidad de i[t]eraciones dada, uso de [m]emoria y parámetros de [p]aralelismo dados:

```
echo "{{contraseña}}" | argon2 "{{texto_sal}}" -t {{5}} -m {{20}} -p {{7}}
```

# aria2

Este comando es un alias de **aria2c**.

- Muestra documentación para el comando actualizado:

```
tldr aria2c
```

# aria2c

Utilidad de descarga rápida.

Soporta HTTP(S), FTP, SFTP, BitTorrent y Metalink.

Más información: <https://aria2.github.io>.

- Descarga un URI específico a un archivo:

```
aria2c "{{url}}"
```

- Descarga un archivo de una URI con un nombre de salida específico:

```
aria2c --out {{ruta/al/archivo}} "{{url}}"
```

- Descarga varios archivos diferentes en paralelo:

```
aria2c --force-sequential {{false}} "{{url1 url2 ...}}"
```

- Descarga el mismo archivo desde diferentes espejos y verifica la suma de comprobación del archivo descargado:

```
aria2c --checksum {{sha-256}}={{suma_de_comprobación}}  
"{{url1}}" "{{url2}}" "{{urlN}}"
```

- Descarga las URI enumeradas en un archivo con un número determinado de descargas paralelas:

```
aria2c --input-file {{ruta/al/archivo}} --max-concurrent-  
downloads {{número_de_descargas}}
```

- Descarga con varias conexiones:

```
aria2c --split {{número_de_conexiones}} "{{url}}"
```

- Descarga a través de FTP con un nombre de usuario y contraseña:

```
aria2c --ftp-user {{nombre_usuario}} --ftp-passwd  
{{contraseña}} "{{url}}"
```

- Limita la velocidad de descarga en bytes por segundo:

```
aria2c --max-download-limit {{velocidad}} "{{url}}"
```

# arp

Muestra y manipula la caché ARP del sistema.

Más información: <https://manned.org/arp>.

- Muestra la tabla ARP actual:

```
arp -a
```

- Borra una entrada específica:

```
arp -d {{dirección}}
```

- Añade una nueva entrada en la tabla ARP:

```
arp -s {{dirección}} {{dirección_mac}}
```

# arping

Descubre y sondea hosts en una red utilizando el protocolo ARP.

Útil para el descubrimiento de direcciones MAC.

Más información: <https://github.com/ThomasHabets/arping>.

- Haz ping a un host mediante paquetes de petición ARP:

```
arping {{host_ip}}
```

- Haz ping a un host en una interfaz específica:

```
arping -I {{interfaz}} {{host_ip}}
```

- Haz ping a un host y detenerse en la primera respuesta:

```
arping -f {{host_ip}}
```

- Haz ping a un host un determinado número de veces:

```
arping -c {{cuenta}} {{host_ip}}
```

- Emite paquetes de solicitud ARP para actualizar las cachés ARP de los vecinos:

```
arping -U {{ip_a_retransmitir}}
```

- Detecta direcciones IP duplicadas en la red enviando peticiones ARP con un tiempo de espera de 3 segundos:

```
arping -D -w {{3}} {{ip_a_verificar}}
```

# asar

Un archivador de ficheros para la plataforma Electron.

Más información: <https://github.com/electron/asar>.

- Archiva un fichero o directorio:

```
asar pack {{ruta/al/archivo_o_directorio}} {{archivado.asar}}
```

- Extrae un archivo:

```
asar extract {{archivado.asar}}
```

- Extrae un archivo específico de un archivo:

```
asar extract-file {{archivado.asar}} {{archivo}}
```

- Lista el contenido de un archivo:

```
asar list {{archivado.asar}}
```

# asciidoctor

Un procesador que convierte archivos AsciiDoc a un formato publicable.

Más información: <https://docs.asciidoctor.org>.

- Convierte un archivo .adoc específico a HTML (el formato de salida por defecto):

```
asciidoctor {{ruta/al/archivo.adoc}}
```

- Convierte un archivo .adoc específico a HTML y vincula una hoja de estilos CSS:

```
asciidoctor -a stylesheet {{ruta/al/stylesheet.css}} {{ruta/al/archivo.adoc}}
```

- Convierte un archivo específico .adoc en HTML incrustable, eliminando todo excepto el cuerpo:

```
asciidoctor --embedded {{ruta/al/archivo.adoc}}
```

- Convierte un archivo .adoc dado en un PDF utilizando la biblioteca asciidoctor-pdf:

```
asciidoctor --backend {{pdf}} --require {{asciidoctor-pdf}}  
{{ruta/al/archivo.adoc}}
```



# asciinema

Graba y reproduce sesiones de terminal, y opcionalmente compártelas en [asciinema.org](https://asciinema.org).

Más información: <https://docs.asciinema.org/manual/cli/usage>.

- Asocia el programa local de `asciinema` con una cuenta de `asciinema.org`:

```
asciinema auth
```

- Crea una nueva grabación (una vez acabada, se preguntará al usuario si la quiere guardar en local, o subirla):

```
asciinema rec
```

- Crea una nueva grabación y la guarda en un archivo local:

```
asciinema rec {{ruta/hacia/grabación.cast}}
```

- Reproduce una grabación desde un archivo local:

```
asciinema play {{ruta/hacia/grabación.cast}}
```

- Reproduce una grabación desde `asciinema.org`:

```
asciinema play https://asciinema.org/a/  
{{identificador_de_grabación}}
```

- Crea una nueva grabación, limitando el tiempo de espera máximo a 2.5 segundos:

```
asciinema rec {{-i|--idle-time-limit}} 2.5
```

- Imprime la salida completa de un archivo local de grabación:

```
asciinema cat {{ruta/hacia/grabación.cast}}
```

- Sube un archivo local de grabación a `asciinema.org`:

```
asciinema upload {{ruta/hacia/grabación.cast}}
```

# aspell

Corrector ortográfico interactivo.

Más información: <http://aspell.net/>.

- Revisa la ortografía de un solo archivo:

```
aspell check {{ruta/al/archivo}}
```

- Lista las palabras mal escritas de `stdin`:

```
cat {{ruta/al/archivo}} | aspell list
```

- Muestra los idiomas disponibles en el diccionario:

```
aspell dicts
```

- Ejecuta `aspell` con un idioma diferente (toma el código de idioma ISO 639 de dos letras):

```
aspell --lang={{cs}}
```

- Lista las palabras mal escritas de `stdin` e ignora las palabras de la lista personal de palabras:

```
cat {{ruta/al/archivo}} | aspell --  
personal={{lista_personal_de_palabras.pws}} list
```

# assimp

Cliente de línea de comandos para la biblioteca Open Asset Import.

Admite la carga de más de 40 formatos de archivo 3D y la exportación a varios formatos 3D populares.

Más información: <https://assimp-docs.readthedocs.io/>.

- Lista todos los formatos de importación soportados:

```
assimp listext
```

- Lista todos los formatos de exportación compatibles:

```
assimp listexport
```

- Convierte un archivo a uno de los formatos de salida soportados, utilizando los parámetros por defecto:

```
assimp export {{archivo_entrada.stl}} {{archivo_salida.obj}}
```

- Convierte un archivo utilizando parámetros personalizados (el archivo `dox_cmd.h` en el código fuente de assimp enumera los parámetros disponibles):

```
assimp export {{archivo_entrada.stl}} {{archivo_salida.obj}}  
{{parametros}}
```

- Muestra un resumen del contenido de un archivo 3D:

```
assimp info {{ruta/al/archivo}}
```

- Lista todos los subcomandos ("verbs") soportados:

```
assimp help
```

- Muestra información de ayuda sobre un subcomando concreto (por ejemplo, los parámetros específicos del mismo):

```
assimp {{subcomando}} --help
```

# astronomer

Herramienta que detecta estrellas ilegítimas de cuentas bot en proyectos de GitHub.

Más información: <https://github.com/Ullaakut/astronomer>.

- Escanea un repositorio:

```
astronomer {{tldr-pages/tldr-node-client}}
```

- Escanea el máximo de estrellas del repositorio:

```
astronomer {{tldr-pages/tldr-node-client}} --stars {{50}}
```

- Escanea un repositorio incluyendo informes comparativos:

```
astronomer {{tldr-pages/tldr-node-client}} --verbose
```

# astyle

Indentador, formateador y embellecedor de código fuente para los lenguajes de programación C, C++, C# y Java.

Al ejecutarse, se crea una copia del archivo original con un ".orig" añadido al nombre del archivo original.

Más información: <https://astyle.sourceforge.net>.

- Aplica el estilo por defecto de 4 espacios por sangría y sin cambios de formato:

```
astyle {{archivo_de_origen}}
```

- Aplica el estilo Java con llaves adjuntas:

```
astyle --style=java {{ruta/al/archivo}}
```

- Aplica el estilo allman con llaves discontinuas:

```
astyle --style=allman {{ruta/al/archivo}}
```

- Aplica una sangría personalizada utilizando espacios. Elige entre 2 y 20 espacios:

```
astyle --indent=spaces={{número_de_espacios}} {{ruta/al/archivo}}
```

- Aplica una sangría personalizada utilizando tabuladores. Elige entre 2 y 20 tabulaciones:

```
astyle --indent=tab={{número_de_pestañas}} {{ruta/al/archivo}}
```

# at

Ejecuta comandos una vez en un momento posterior.

El servicio atd (o atrun) debe estar ejecutándose para las ejecuciones reales.

Más información: <https://manned.org/at>.

- Ejecuta comandos desde la entrada estándar en 5 minutos (pulsa `Ctrl + D` cuando termines):

```
at now + 5 minutes
```

- Ejecuta un comando desde la entrada estándar a las 10:00 AM de hoy:

```
echo "{{./make_db_backup.sh}}" | at 1000
```

- Ejecuta comandos desde un archivo dado el próximo martes:

```
at -f {{ruta/al/archivo}} 9:30 PM Tue
```

# atoum

Un framework de pruebas unitarias para PHP sencillo, moderno e intuitivo.

Más información: <https://atoum.org>.

- Inicializa un fichero de configuración:

```
atoum --init
```

- Ejecuta todas las pruebas:

```
atoum
```

- Ejecuta pruebas utilizando el archivo de configuración especificado:

```
atoum -c {{ruta/al/archivo}}
```

- Ejecuta un archivo de prueba específico:

```
atoum -f {{ruta/al/archivo}}
```

- Ejecuta un directorio específico de pruebas:

```
atoum -d {{ruta/al/directorio}}
```

- Ejecuta todas las pruebas dado un namespace específico:

```
atoum -ns {{namespace}}
```

- Ejecuta todas las pruebas dada una etiqueta específica:

```
atoum -t {{etiqueta}}
```

- Carga un archivo bootstrap personalizado antes de ejecutar las pruebas:

```
atoum --bootstrap-file {{ruta/al/archivo}}
```

# audacious

Un reproductor de audio de código abierto.

Más información: <https://audacious-media-player.org>.

- Inicia el programa:

```
audacious
```

- Pone en cola un directorio específico de archivos de audio:

```
audacious --enqueue {{ruta/al/directorio}}
```

- Inicia o detiene la reproducción:

```
audacious --play-pause
```

- Avanza o retrocede en la lista de reproducción:

```
audacious --{{fwd|rew}}
```

- Detiene la reproducción:

```
audacious --stop
```

- Inicia una versión acéfala:

```
audacious --headless
```

- Sale en cuanto se detiene la reproducción o mientras no haya nada que reproducir:

```
audacious --quit-after-play
```



# audtool

Controla Audacious usando comandos.

Más información: <https://manned.org/audtool>.

- Reproduce/pausa la reproducción de audio:

```
audtool playback-playpause
```

- Imprime el nombre del artista, el álbum y la canción que se está reproduciendo:

```
audtool current-song
```

- Ajusta el volumen de la reproducción de audio:

```
audtool set-volume {{100}}
```

- Salta a la siguiente canción:

```
audtool playlist-advance
```

- Imprime la tasa de bits de la canción actual en kilobits:

```
audtool current-song-bitrate-kbps
```

- Abre Audacious en pantalla completa si está oculto:

```
audtool mainwin-show
```

- Muestra ayuda:

```
audtool help
```

- Muestra configuración:

```
audtool preferences-show
```

# autoflake

Una herramienta para eliminar importaciones y variables no utilizadas del código Python.

Más información: <https://github.com/myint/autoflake>.

- Elimina las variables no utilizadas de un archivo y muestra la diferencia:

```
autoflake --remove-unused-variables {{ruta/al/archivo.py}}
```

- Elimina las importaciones no utilizadas de varios archivos y muestra las diferencias:

```
autoflake --remove-all-unused-imports {{ruta/al/archivo1.py  
ruta/al/archivo2.py ...}}
```

- Elimina variables no utilizadas de un fichero, sobrescribiendo el fichero:

```
autoflake --remove-unused-variables --in-place {{ruta/al/  
archivo.py}}
```

- Elimina recursivamente las variables no utilizadas de todos los archivos de un directorio, sobrescribiendo cada archivo:

```
autoflake --remove-unused-variables --in-place --recursive  
{{ruta/al/directorio}}
```

# autopep8

Formatea el código Python según la guía de estilo PEP 8.

Más información: <https://github.com/hhatto/autopep8>.

- Formatea un archivo a `stdout`, con una longitud de línea máxima personalizada:

```
autopep8 {{ruta/al/archivo.py}} --max-line-length {{length}}
```

- Formatea un fichero, mostrando un diff de los cambios:

```
autopep8 --diff {{ruta/al/archivo}}
```

- Formatea un fichero en su lugar y guarda los cambios:

```
autopep8 --in-place {{ruta/al/archivo.py}}
```

- Formatea recursivamente todos los archivos de un directorio y guarda los cambios:

```
autopep8 --in-place --recursive {{ruta/al/directorio}}
```

# autossh

Ejecuta, monitorea y reinicia conexiones SSH.

Auto-reconecta para mantener los túneles de reenvío de puertos. Acepta todas las señales SSH.

Más información: <https://www.harding.motd.ca/autossh>.

- Inicia una sesión SSH, reiniciando cuando un puerto de monitoreo no retorna datos:

```
autossh -M {{puerto_monitor}} "{{comando_ssh}}"
```

- Reenvía un puerto local a uno remoto, reiniciando cuando sea necesario:

```
autossh -M {{puerto_monitor}} -L {{puerto_local}}:localhost:{{puerto_remoto}} {{usuario}}@{{host}}
```

- Crea un proceso autossh en segundo plano antes de ejecutar SSH y no abre un shell remoto:

```
autossh -f -M {{puerto_monitor}} -N "{{comando_ssh}}"
```

- Ejecuta en segundo plano, sin puerto de monitorización, y en su lugar envía paquetes SSH keep-alive cada 10 segundos para detectar fallos:

```
autossh -f -M 0 -N -o "ServerAliveInterval 10" -o "ServerAliveCountMax 3" "{{comando_ssh}}"
```

- Ejecuta en segundo plano, sin puerto de monitorización y sin shell remoto, saliendo si falla el reenvío de puerto:

```
autossh -f -M 0 -N -o "ServerAliveInterval 10" -o "ServerAliveCountMax 3" -o ExitOnForwardFailure=yes -L {{local_port}}:localhost:{{puerto_remoto}} {{usuario}}@{{host}}
```

- Se ejecuta en segundo plano, registrando la salida de depuración autossh y la salida detallada SSH en archivos:

```
AUTOSSH_DEBUG=1 AUTOSSH_LOGFILE={{ruta/al/autossh_log_file.log}} autossh -f -M {{puerto_monitor}} -v -E {{ruta/al/archivo_ssh_log.log}} {{comando_ssh}}
```

# avo

La interfaz oficial de línea de comandos para Avo.

Más información: <https://www.avo.app/docs/implementation/cli>.

- Inicializa un espacio de trabajo en el directorio actual:

```
avo init
```

- Inicia sesión en la plataforma Avo:

```
avo login
```

- Cambia a una rama Avo existente:

```
avo checkout {{nombre_rama}}
```

- Extrae las envolturas analíticas de la ruta actual:

```
avo pull
```

- Muestra el estado de la implementación de Avo:

```
avo status
```

- Resuelve conflictos Git en archivos Avo:

```
avo conflict
```

- Abre el espacio de trabajo actual de Avo en el navegador web predeterminado:

```
avo edit
```

- Muestra la ayuda de un subcomando:

```
avo {{subcomando}} --help
```

# avrdude

Programa controlador para la programación de microcontroladores Atmel AVR.

Más información: <https://www.nongnu.org/avrdude/>.

- Lee el microcontrolador AVR:

```
avrdude -p {{AVR_dispositivo}} -c {{programador}} -U flash:r:{{file.hex}}:i
```

- Escribe el microcontrolador AVR:

```
avrdude -p {{AVR_dispositivo}} -c {{programador}} -U flash:w:{{file.hex}}
```

- Lista de dispositivos AVR disponibles:

```
avrdude -p \?
```

- Lista de programadores AVR disponibles:

```
avrdude -c \?
```

# awk

Un versátil lenguaje de programación para trabajar con archivos.

Más información: <https://github.com/onetrueawk/awk>.

- Imprime la quinta columna (también conocida como campo) de un archivo separado por espacios:

```
awk '{print $5}' {{ruta/al/archivo}}
```

- Imprime la segunda columna de las líneas que contienen "foo" en un archivo separado por espacios:

```
awk '/{{foo}}/ {print $2}' {{ruta/al/archivo}}
```

- Imprime la última columna de cada línea de un archivo, utilizando una coma (en lugar de un espacio) como separador de campos:

```
awk -F ',' '{print $NF}' {{ruta/al/archivo}}
```

- Suma los valores de la primera columna de un fichero e imprime el total:

```
awk '{s+=$1} END {print s}' {{ruta/al/archivo}}
```

- Imprime una de cada tres líneas a partir de la primera:

```
awk 'NR%3==1' {{ruta/al/archivo}}
```

- Imprime diferentes valores basados en condiciones:

```
awk '{if ($1 == "foo") print "Coincidencia exacta foo"; else if ($1 ~ "bar") print "Coincidencia parcial bar"; else print "Baz"}' {{ruta/al/archivo}}
```

- Imprime todas las líneas en las que el valor de la 10ª columna está entre un mínimo y un máximo:

```
awk '($10 >= {{valor_mínimo}} && $10 <= {{valor_máximo}})'
```

- Imprime tabla de usuarios con UID >=1000 con cabecera y salida formateada, usando dos puntos como separador (%-20s significa: 20 caracteres de cadena alineados a la izquierda, %6s significa: 6 caracteres de cadena alineados a la derecha):

```
awk 'BEGIN {FS=":";printf "%-20s %6s %25s\n", "Name", "UID", "Shell"} $4 >= 1000 {printf "%-20s %6d %25s\n", $1, $4, $7}' /etc/passwd
```

# aws accessanalyzer

Analiza y revisa las políticas de recursos para identificar posibles riesgos de seguridad.

Más información: <https://awscli.amazonaws.com/v2/documentation/api/latest/reference/accessanalyzer/index.html>.

- Crea un nuevo Access Analyzer:

```
aws accessanalyzer create-analyzer --analyzer-name  
{{nombre_analizador}} --type {{tipo}} --tags {{etiquetas}}
```

- Elimina un analizador de acceso existente:

```
aws accessanalyzer delete-analyzer --analyzer-arn  
{{analizador_arn}}
```

- Obtiene detalles de un analizador de acceso específico:

```
aws accessanalyzer get-analyzer --analyzer-arn  
{{analizador_arn}}
```

- Lista todos los analizadores de acceso:

```
aws accessanalyzer list-analyzers
```

- Actualiza la configuración de un analizador de acceso:

```
aws accessanalyzer update-analyzer --analyzer-arn  
{{analizador_arn}} --tags {{nuevas_etiquetas}}
```

- Crea una nueva regla de archivo de Access Analyzer:

```
aws accessanalyzer create-archive-rule --analyzer-arn  
{{analizador_arn}} --rule-name {{nombre_regla}} --filter  
{{filtro}}
```

- Elimina una regla de archivo de Access Analyzer:

```
aws accessanalyzer delete-archive-rule --analyzer-arn  
{{analizador_arn}} --rule-name {{nombre_regla}}
```

- Lista todas las reglas de archivo de Access Analyzer:

```
aws accessanalyzer list-archive-rules --analyzer-arn  
{{analizador_arn}}
```



# aws acm

AWS Certificate Manager.

Más información: <https://awscli.amazonaws.com/v2/documentation/api/latest/reference/acm/index.html>.

- Importa un certificado:

```
aws acm import-certificate --certificate-arn  
{{certificado_arn}} --certificate {{certificado}} --private-  
key {{clave_privada}} --certificate-chain  
{{certificado_cadena}}
```

- Lista certificados:

```
aws acm list-certificates
```

- Describe un certificado:

```
aws acm describe-certificate --certificate-arn  
{{certificado_arn}}
```

- Solicita un certificado:

```
aws acm request-certificate --domain-name {{nombre_dominio}}  
--validation-method {{método_de_validación}}
```

- Elimina un certificado:

```
aws acm delete-certificate --certificate-arn  
{{certificate_arn}}
```

- Lista validaciones de certificados:

```
aws acm list-certificates --certificate-statuses {{estado}}
```

- Obtén detalles del certificado:

```
aws acm get-certificate --certificate-arn {{certificado_arn}}
```

- Actualiza las opciones del certificado:

```
aws acm update-certificate-options --certificate-arn  
{{certificado_arn}} --options {{opciones}}
```

# aws amplify

Plataforma de desarrollo para crear aplicaciones móviles y web seguras y escalables.

Más información: <https://awscli.amazonaws.com/v2/documentation/api/latest/reference/amplify/index.html>.

- Crea una nueva aplicación Amplify:

```
aws amplify create-app --name {{nombre_de_la_app}} --  
description {{descripción}} --repository {{url_repositorio}}  
--platform {{plataforma}} --environment-variables  
{{variables_de_entorno}} --tags {{etiquetas}}
```

- Elimina una aplicación Amplify existente:

```
aws amplify delete-app --app-id {{id_aplicación}}
```

- Obtiene detalles de una aplicación Amplify determinada:

```
aws amplify get-app --app-id {{id_aplicación}}
```

- Lista todas las aplicaciones de Amplify:

```
aws amplify list-apps
```

- Actualiza la configuración de una aplicación Amplify:

```
aws amplify update-app --app-id {{id_aplicación}} --name  
{{nuevo_nombre}} --description {{nueva_descripción}} --  
repository {{nuevo_url_repositorio}} --environment-variables  
{{variables_nuevo_entorno}} --tags {{nuevas_etiquetas}}
```

- Añade un nuevo entorno backend a una aplicación Amplify:

```
aws amplify create-backend-environment --app-id  
{{id_aplicación}} --environment-name {{nombre_del_entorno}}  
--deployment-artifacts {{artefactos}}
```

- Elimina un entorno backend de una aplicación Amplify:

```
aws amplify delete-backend-environment --app-id  
{{id_aplicación}} --environment-name {{nombre_del_entorno}}
```

- Lista todos los entornos backend de una aplicación Amplify:

```
aws amplify list-backend-environments --app-id  
{{id_aplicación}}
```

# aws ce

Ejecuta operaciones de gestión de costos a través del servicio AWS Cost Explorer.

Más información: <https://awscli.amazonaws.com/v2/documentation/api/latest/reference/ce/index.html>.

- Crea monitor de anomalías:

```
aws ce create-anomaly-monitor --monitor {{nombre_monitor}} --monitor-type {{tipo_monitor}}
```

- Crea suscripción de anomalías:

```
aws ce create-anomaly-subscription --subscription {{nombre_de_suscripción}} --monitor-arn {{monitor_arn}} --subscribers {{suscriptores}}
```

- Obtiene anomalías:

```
aws ce get-anomalies --monitor-arn {{monitor_arn}} --start-time {{hora_de_inicio}} --end-time {{hora_final}}
```

- Obtiene coste y uso:

```
aws ce get-cost-and-usage --time-period {{fecha_inicio}}/{{fecha_final}} --granularity {{granularidad}} --metrics {{métricas}}
```

- Obtiene previsión de costes:

```
aws ce get-cost-forecast --time-period {{fecha_inicio}}/{{fecha_final}} --granularity {{granularidad}} --metric {{métrica}}
```

- Obtiene la utilización de la reserva:

```
aws ce get-reservation-utilization --time-period {{fecha_inicio}}/{{fecha_final}} --granularity {{granularidad}}
```

- Lista de definiciones de categorías de costes:

```
aws ce list-cost-category-definitions
```

- Recurso de etiquetas:

```
aws ce tag-resource --resource-arn {{recurso_arn}} --tags {{etiquetas}}
```

# aws codecommit

Un servicio de control de versión capaz de alojar repositorios de Git privados.

Más información: <https://docs.aws.amazon.com/cli/latest/reference/codecommit/>.

- Muestra ayuda:

```
aws codecommit help
```

- Muestra ayuda de un comando:

```
aws codecommit {{comando}} help
```

# aws cognito-idp

Administra el grupo de usuarios de Amazon Cognito y sus usuarios y grupos utilizando la CLI.

Más información: <https://awscli.amazonaws.com/v2/documentation/api/latest/reference/cognito-idp/index.html>.

- Crea un nuevo grupo de usuarios de Cognito:

```
aws cognito-idp create-user-pool --pool-name {{nombre}}
```

- Lista todos los grupos de usuarios:

```
aws cognito-idp list-user-pools --max-results {{10}}
```

- Elimina un grupo de usuarios específico:

```
aws cognito-idp delete-user-pool --user-pool-id  
{{identificador_de_pool}}
```

- Crea un usuario en un grupo específico:

```
aws cognito-idp admin-create-user --username  
{{nombre_usuario}} --user-pool-id {{identificador_de_pool}}
```

- Lista los usuarios de un pool específico:

```
aws cognito-idp list-users --user-pool-id  
{{identificador_de_pool}}
```

- Elimina un usuario de un grupo específico:

```
aws cognito-idp admin-delete-user --username  
{{nombre_usuario}} --user-pool-id {{identificador_de_pool}}
```

# aws configure

Gestiona la configuración para la CLI de AWS.

Más información: <https://docs.aws.amazon.com/cli/latest/reference/configure/>.

- Configura AWS CLI interactivamente (crea una nueva configuración o actualiza la predeterminada):

```
aws configure
```

- Configura un perfil con nombre para la CLI de AWS de forma interactiva (crea un perfil nuevo o actualiza uno existente):

```
aws configure --profile {{nombre_del_perfil}}
```

- Muestra el valor de una variable de configuración específica:

```
aws configure get {{nombre}}
```

- Muestra el valor de una variable de configuración en un perfil específico:

```
aws configure get {{nombre}} --profile {{nombre_del_perfil}}
```

- Establece el valor de una variable de configuración específica:

```
aws configure set {{nombre}} {{valor}}
```

- Establece el valor de una variable de configuración en un perfil específico:

```
aws configure set {{nombre}} {{valor}} --profile {{nombre_del_perfil}}
```

- Lista las entradas de configuración:

```
aws configure list
```

- Lista las entradas de configuración para un perfil específico:

```
aws configure list --profile {{nombre_del_perfil}}
```

# aws cur

Crea, solicita y elimina definiciones de informes de uso de AWS.

Más información: <https://awscli.amazonaws.com/v2/documentation/api/latest/reference/cur/index.html>.

- Crea una definición de informe de costes y uso de AWS a partir de un archivo JSON:

```
aws cur put-report-definition --report-definition file://  
{{ruta/al/report_definition.json}}
```

- Enumera las definiciones de informes de uso definidas para la cuenta conectada:

```
aws cur describe-report-definitions
```

- Elimina una definición de informe de uso:

```
aws cur --region {{aws_region}} delete-report-definition --  
report-name {{report}}
```

# aws ec2

Interfaz de línea de comandos (CLI) para AWS EC2.

Provee capacidad de computación segura y redimensionable en la nube de AWS, permitiendo mayor velocidad en el desarrollo e implementación de aplicaciones.

Más información: <https://awscli.amazonaws.com/v2/documentation/api/latest/reference/ec2/index.html>.

- Muestra información acerca de una instancia específica:

```
aws ec2 describe-instances --instance-ids  
{{identificador_de_instancia}}
```

- Muestra información sobre todas las instancias:

```
aws ec2 describe-instances
```

- Muestra información sobre todos los volúmenes EC2:

```
aws ec2 describe-volumes
```

- Elimina un volumen EC2:

```
aws ec2 delete-volume --volume-id  
{{identificador_de_volumen}}
```

- Crea una instantánea a partir de un volumen EC2:

```
aws ec2 create-snapshot --volume-id  
{{identificador_de_volumen}}
```

- Lista las imágenes de máquina de Amazon disponibles (AMI):

```
aws ec2 describe-images
```

- Lista todos los comandos EC2 disponibles:

```
aws ec2 help
```

- Muestra la ayuda para un comando EC2 específico:

```
aws ec2 {{subcomando}} help
```



# aws glue

CLI para AWS Glue.

Define el punto de enlace público para el servicio AWS Glue.

Más información: <https://docs.aws.amazon.com/cli/latest/reference/glue/>.

- Lista trabajos:

```
aws glue list-jobs
```

- Inicia un trabajo:

```
aws glue start-job-run --job-name {{nombre_del_trabajo}}
```

- Inicia la ejecución de un flujo de trabajo:

```
aws glue start-workflow-run --name {{nombre_del_flujo}}
```

- Lista disparadores:

```
aws glue list-triggers
```

- Inicia un disparador:

```
aws glue start-trigger --name {{nombre_disparador}}
```

- Crea un punto final de desarrollo:

```
aws glue create-dev-endpoint --endpoint-name {{nombre}} --  
role-arn {{role_arn_usado_por_puntofinal}}
```

# aws-google-auth

Herramienta de línea de comandos para adquirir credenciales temporales de AWS (STS) utilizando Google Apps como proveedor federado (Single Sign-On).

Más información: <https://github.com/cevoaustralia/aws-google-auth>.

- Inicia sesión con Google SSO utilizando los identificadores IDP y SP y establece la duración de las credenciales en una hora:

```
aws-google-auth -u {{ejemplo@ejemplo.com}} -I  
{{${G00GLE_IDP_ID}} -S {{${G00GLE_SP_ID}} -d {{3600}}
```

- Inicia sesión preguntando qué rol usar (en caso de varios roles disponibles SAML):

```
aws-google-auth -u {{ejemplo@ejemplo.com}} -I  
{{${G00GLE_IDP_ID}} -S {{${G00GLE_SP_ID}} -d {{3600}} -a
```

- Resuelve alias para cuentas AWS:

```
aws-google-auth -u {{ejemplo@ejemplo.com}} -I  
{{${G00GLE_IDP_ID}} -S {{${G00GLE_SP_ID}} -d {{3600}} -a --  
resolve-aliases
```

- Muestra información de ayuda:

```
aws-google-auth -h
```

# aws help

Muestra información de ayuda sobre la CLI de AWS.

Más información: <https://docs.aws.amazon.com/cli/latest/userguide/cli-usage-help.html>.

- Muestra la ayuda:

```
aws help
```

- Lista todos los temas disponibles:

```
aws help topics
```

- Muestra ayuda sobre un tema específico:

```
aws help {{nombre_tema}}
```

# aws iam

Interactúa con el Manejo de Identidad y Acceso (o "IAM" en inglés), un servicio web para controlar seguramente el acceso a servicios de AWS.

Más información: <https://awscli.amazonaws.com/v2/documentation/api/latest/reference/iam/index.html>.

- Lista usuarios:

```
aws iam list-users
```

- Lista políticas:

```
aws iam list-policies
```

- Lista grupos:

```
aws iam list-groups
```

- Obtén los usuarios en un grupo:

```
aws iam get-group --group-name {{nombre_del_grupo}}
```

- Describe una política IAM:

```
aws iam get-policy --policy-arn arn:aws:iam::aws:policy/  
{{nombre_de_política}}
```

- Lista claves de acceso:

```
aws iam list-access-keys
```

- Lista claves de acceso para un usuario específico:

```
aws iam list-access-keys --user-name {{nombre_de_usuario}}
```

- Muestra ayuda:

```
aws iam help
```

# aws kendra

CLI para AWS Kendra.

Más información: <https://awscli.amazonaws.com/v2/documentation/api/latest/reference/kendra/index.html>.

- Crea un índice:

```
aws kendra create-index --name {{nombre}} --role-arn  
{{rol_arn}}
```

- Lista índices:

```
aws kendra list-indexes
```

- Describe un índice:

```
aws kendra describe-index --id {{id_índice}}
```

- Lista fuentes de datos:

```
aws kendra list-data-sources
```

- Describe una fuente de datos:

```
aws kendra describe-data-source --id {{id_fuente_datos}}
```

- Lista consultas de búsqueda:

```
aws kendra list-query-suggestions --index-id {{id_índice}} --  
query-text {{texto_consulta}}
```

# aws kinesis

CLI oficial de AWS para los servicios de streaming de datos de Amazon Kinesis.

Más información: <https://docs.aws.amazon.com/cli/latest/reference/kinesis/index.html#cli-aws-kinesis>.

- Muestra todos los streams de la cuenta:

```
aws kinesis list-streams
```

- Escribe un registro en un flujo de Kinesis:

```
aws kinesis put-record --stream-name {{nombre}} --partition-key {{clave}} --data {{base64_encoded_message}}
```

- Escribe un registro en un flujo Kinesis con codificación base64 en línea:

```
aws kinesis put-record --stream-name {{nombre}} --partition-key {{clave}} --data "$( echo "{{my raw message}}" | base64 )"
```

- Lista los fragmentos disponibles en un flujo:

```
aws kinesis list-shards --stream-name {{nombre}}
```

- Obtén un iterador de fragmentos para leer el mensaje más antiguo de un fragmento de flujo:

```
aws kinesis get-shard-iterator --shard-iterator-type TRIM_HORIZON --stream-name {{nombre}} --shard-id {{id}}
```

- Lee registros de un fragmento utilizando un iterador de fragmento:

```
aws kinesis get-records --shard-iterator {{iterador}}
```

# aws pricing

Consulta servicios, productos e información de precios de Amazon Web Services.

Más información: <https://docs.aws.amazon.com/cli/latest/reference/pricing/>.

- Lista códigos de servicio de una región específica:

```
aws pricing describe-services --region {{us-east-1}}
```

- Lista atributos para un código de servicio dado en una región específica:

```
aws pricing describe-services --service-code {{AmazonEC2}} --  
region {{us-east-1}}
```

- Imprime información de precios para un código de servicio en una región específica:

```
aws pricing get-products --service-code {{AmazonEC2}} --  
region {{us-east-1}}
```

- Lista valores para un atributo específico para un código de servicio en una región específica:

```
aws pricing get-attribute-values --service-code {{AmazonEC2}}  
--attribute-name {{instanceType}} --region {{us-east-1}}
```

- Imprime información de precios para un código de servicio usando filtros por tipo de instancia y ubicación:

```
aws pricing get-products --service-code {{AmazonEC2}} --  
filters  
"{{Type=TERM_MATCH,Field=instanceType,Value=m5.xlarge}}"  
"{{Type=TERM_MATCH,Field=location,Value=US East (N.  
Virginia)}}"
```

# aws rds

CLI para AWS Relational Database Service.

Crea y administra bases de datos relacionales.

Más información: <https://awscli.amazonaws.com/v2/documentation/api/latest/reference/rds/index.html>.

- Muestra ayuda para subcomando RDS específicos:

```
aws rds {{subcommand}} help
```

- Detiene instancia:

```
aws rds stop-db-instance --db-instance-identifier  
{{identificador_de_instancia}}
```

- Inicia instancia:

```
aws rds start-db-instance --db-instance-identifier  
{{identificador_de_instancia}}
```

- Modifica una instancia RDS:

```
aws rds modify-db-instance --db-instance-identifier  
{{identificador_de_instancia}} {{parametros}} --apply-  
immediately
```

- Aplica actualizaciones a una instancia RDS:

```
aws rds apply-pending-maintenance-action --resource-  
identifier {{database_arn}} --apply-action {{system-update}}  
--opt-in-type {{immediate}}
```

- Modifica un identificador de instancia:

```
aws rds modify-db-instance --db-instance-identifier  
{{antiguo_identificador_instancia}} --new-db-instance-  
identifier {{nuevo_identificador_instancia}}
```

- Reinicia una instancia:

```
aws rds reboot-db-instance --db-instance-identifier  
{{identificador_de_instancia}}
```

- Elimina una instancia:



```
aws rds delete-db-instance --db-instance-identifier  
{{identificador_de_instancia}} --final-db-snapshot-identifier  
{{identificador_snapshot}} --delete-automated-backups
```

# aws route53

CLI para AWS Route53 - Route 53 es un servicio web de Sistema de Nombres de Dominio (DNS) altamente disponible y escalable.

Más información: <https://awscli.amazonaws.com/v2/documentation/api/latest/reference/route53/index.html>.

- Lista todas las zonas alojadas, privadas y públicas:

```
aws route53 list-hosted-zones
```

- Muestra todos los registros de una zona:

```
aws route53 list-resource-record-sets --hosted-zone-id  
{{identificador_de_zona}}
```

- Crea una nueva zona pública utilizando un identificador de solicitud para reintentar la operación de forma segura:

```
aws route53 create-hosted-zone --name {{nombre}} --caller-  
reference {{identificador_de_solicitud}}
```

- Elimina una zona (si la zona tiene registros SOA y NS no predeterminados, el comando fallará):

```
aws route53 delete-hosted-zone --id {{identificador_de_zona}}
```

- Prueba la resolución DNS por parte de los servidores de Amazon de una zona determinada:

```
aws route53 test-dns-answer --hosted-zone-id  
{{identificador_de_zona}} --record-name {{nombre}} --record-  
type {{tipo}}
```

# axel

Acelerador de descargas.

Protocolos soportados HTTP, HTTPS y FTP.

Más información: <https://github.com/axel-download-accelerator/axel>.

- Descarga un archivo alojado en una URL:

```
axel {{url}}
```

- Descarga y especifica un nombre de archivo:

```
axel {{url}} -o {{ruta/al/archivo}}
```

- Descarga con múltiples conexiones:

```
axel -n {{num_conexiones}} {{url}}
```

- Busca copias espejo:

```
axel -S {{num_de_espejos}} {{url}}
```

- Limita la velocidad de descarga (bytes por segundo):

```
axel -s {{velocidad}} {{url}}
```

# az account

Administra la información de una suscripción de Azure.

Parte de **azure-cli** (también conocido como **az**).

Más información: <https://learn.microsoft.com/cli/azure/account>.

- Lista las suscripciones de la cuenta activa:

```
az account list
```

- Establece una `subscription` como la suscripción activa:

```
az account set --subscription  
{{identificador_de_suscripción}}
```

- Lista las regiones admitidas para la suscripción activa:

```
az account list-locations
```

- Imprime un token de acceso para usar con la MS Graph API:

```
az account get-access-token --resource-type {{ms-graph}}
```

- Imprime los detalles de la suscripción activa actual en un formato específico:

```
az account show --output {{json|tsv|table|yaml}}
```

# az apim

Administra los servicios de Azure API Management.

Parte de **azure-cli** (también conocido como **az**).

Más información: <https://learn.microsoft.com/cli/azure/apim>.

- Enumera las instancias del servicio API Management:

```
az apim list --resource-group {{grupo_de_recursos}}
```

- Crea una instancia de servicio de API Management:

```
az apim create --name {{nombre}} --resource-group  
{{grupo_de_recursos}} --publisher-email {{email}} --  
publisher-name {{name}}
```

- Elimina una instancia del servicio de API Management:

```
az apim delete --name {{nombre}} --resource-group  
{{grupo_de_recursos}}
```

- Muestra detalles de una instancia del servicio de API Management:

```
az apim show --name {{nombre}} --resource-group  
{{grupo_de_recursos}}
```

- Actualiza una instancia del servicio API Management:

```
az apim update --name {{nombre}} --resource-group  
{{grupo_de_recursos}}
```

# az appconfig

Administra las configuraciones de aplicaciones en Azure.

Parte de **azure-cli** (también conocido como **az**).

Más información: <https://learn.microsoft.com/cli/azure/appconfig>.

- Crea una configuración de aplicación:

```
az appconfig create --name {{nombre}} --resource-group  
{{grupo_de_recursos}} --location {{ubicación}}
```

- Elimina una configuración de aplicación específica:

```
az appconfig delete --resource-group {{grupo_de_recursos}} --  
name {{nombre_de_configuración}}
```

- Lista todas las configuraciones de aplicaciones bajo la suscripción actual:

```
az appconfig list
```

- Lista todas las configuraciones de aplicaciones bajo un grupo de recursos específico:

```
az appconfig list --resource-group {{grupo_de_recursos}}
```

- Muestra las propiedades de una configuración de aplicación:

```
az appconfig show --name {{nombre_de_configuración}}
```

- Actualiza una configuración de aplicación específica:

```
az appconfig update --resource-group {{grupo_de_recursos}} --  
name {{nombre_de_configuración}}
```

# az bicep

Grupo de comandos de la CLI de Bicep.

Parte de **azure-cli** (también conocido como **az**).

Más información: <https://learn.microsoft.com/cli/azure/bicep>.

- Instala la CLI de Bicep:

```
az bicep install
```

- Crea un archivo de Bicep:

```
az bicep build --file {{ruta/al/archivo.bicep}}
```

- Intenta descompilar un archivo de plantilla ARM a un archivo de Bicep:

```
az bicep decompile --file {{ruta/al/archivo_plantilla.json}}
```

- Actualiza la CLI de Bicep a la última versión:

```
az bicep upgrade
```

- Muestra la versión instalada de la CLI de Bicep:

```
az bicep version
```

- Lista todas las versiones disponibles de la CLI de Bicep:

```
az bicep list-versions
```

- Desinstala la CLI de Bicep:

```
az bicep uninstall
```

# az config

Administra la configuración de Azure CLI.

Parte de **azure-cli** (también conocido como **az**).

Más información: <https://learn.microsoft.com/cli/azure/config>.

- Muestra todas las configuraciones:

```
az config get
```

- Muestra las configuraciones para una sección específica:

```
az config get {{nombre_de_sección}}
```

- Establece una configuración:

```
az config set {{nombre_de_configuración}}={{valor}}
```

- Elimina una configuración:

```
az config unset {{nombre_de_configuración}}
```



# az devops

Administra organizaciones de Azure DevOps.

Parte de **azure-cli** (también conocido como **az**).

Más información: <https://learn.microsoft.com/cli/azure/devops>.

- Configura el Token de Acceso Personal (PAT) para iniciar sesión en una organización específica:

```
az devops login --organization {{url_de_la_organización}}
```

- Abre un proyecto en el navegador:

```
az devops project show --project {{nombre_de_proyecto}} --open
```

- Lista los miembros de un equipo específico que trabaja en un proyecto en particular:

```
az devops team list-member --project {{nombre_de_proyecto}} --team {{nombre_de_equipo}}
```

- Comprueba la configuración actual de la CLI de Azure DevOps:

```
az devops configure --list
```

- Configura el comportamiento de la CLI de Azure DevOps estableciendo un proyecto predeterminado y una organización predeterminada:

```
az devops configure --defaults project={{nombre_de_proyecto}} organization={{url_de_la_organización}}
```

# az feedback

Envía comentarios al equipo de Azure CLI.

Parte de **azure-cli** (también conocido como **az**).

Más información: [https://learn.microsoft.com/cli/azure/reference-index#az\\_feedback](https://learn.microsoft.com/cli/azure/reference-index#az_feedback).

- Envía comentarios al equipo de Azure CLI:

```
az feedback
```

# az group

Administra grupos de recursos e implementaciones de plantillas.

Parte de **azure-cli** (también conocido como **az**).

Más información: <https://learn.microsoft.com/cli/azure/group>.

- Crea un nuevo grupo de recursos:

```
az group create --name {{nombre}} --location {{ubicación}}
```

- Comprueba si existe un grupo de recursos:

```
az group exists --name {{nombre}}
```

- Elimina un grupo de recursos:

```
az group delete --name {{nombre}}
```

- Coloca un grupo de recursos en estado de espera hasta que se cumpla una condición:

```
az group wait --name {{nombre}} --{{created|deleted|exists|updated}}
```

# az logout

Cierra la sesión de una suscripción de Azure.

Parte de **azure-cli** (también conocido como **az**).

Más información: [https://learn.microsoft.com/cli/azure/reference-index#az\\_logout](https://learn.microsoft.com/cli/azure/reference-index#az_logout).

- Cierra la sesión de la cuenta activa:

```
az logout
```

- Cierra la sesión de un nombre de usuario específico:

```
az logout --username {{alias@somedomain.com}}
```

# az sshkey

Administra claves públicas SSH con máquinas virtuales.

Parte de **azure-cli** (también conocido como **az**).

Más información: <https://learn.microsoft.com/cli/azure/sshkey>.

- Crea una nueva clave SSH:

```
az sshkey create --name {{nombre}} --resource-group  
{{grupo_de_recursos}}
```

- Sube una clave SSH existente:

```
az sshkey create --name {{nombre}} --resource-group  
{{grupo_de_recursos}} --public-key "{{@ruta/de/llave.pub}}"
```

- Lista todas las claves públicas SSH:

```
az sshkey list
```

- Muestra información sobre una clave pública SSH:

```
az sshkey show --name {{nombre}} --resource-group  
{{grupo_de_recursos}}
```

# az storage

Administra los recursos de Azure Cloud Storage.

Parte de **azure-cli** (también conocido como **az**).

Más información: <https://learn.microsoft.com/cli/azure/storage>.

- Crea una cuenta de almacenamiento:

```
az storage account create --resource-group  
{{grupo_de_recursos}} --name {{nombre_de_cuenta}} -l  
{{ubicación}} --sku {{account_sku}}
```

- Enumera todas las cuentas de almacenamiento de un grupo de recursos:

```
az storage account list --resource-group  
{{grupo_de_recursos}}
```

- Enumera las claves de acceso de una cuenta de almacenamiento:

```
az storage account keys list --resource-group  
{{grupo_de_recursos}} --name {{nombre_de_cuenta}}
```

- Elimina una cuenta de almacenamiento:

```
az storage account delete --resource-group  
{{grupo_de_recursos}} --name {{nombre_de_cuenta}}
```

- Actualiza la versión mínima de TLS para una cuenta de almacenamiento:

```
az storage account update --min-tls-version {{TLS1_0|TLS1_1|  
TLS1_2}} --resource-group {{grupo_de_recursos}} --name  
{{nombre_de_cuenta}}
```

# az tag

Administra etiquetas en un recurso de Azure.

Parte de **azure-cli** (también conocido como **az**).

Más información: <https://learn.microsoft.com/cli/azure/tag>.

- Crea un valor de etiqueta:

```
az tag add-value --name {{nombre_de_etiqueta}} --value  
{{valor_de_etiqueta}}
```

- Crea una etiqueta en la suscripción:

```
az tag create --name {{nombre_de_etiqueta}}
```

- Elimina una etiqueta de la suscripción:

```
az tag delete --name {{nombre_de_etiqueta}}
```

- Enumera todas las etiquetas de una suscripción:

```
az tag list --resource-id /subscriptions/  
{{identificador_de_suscripción}}
```

- Elimina un valor de etiqueta para un nombre de etiqueta específico:

```
az tag remove-value --name {{nombre_de_etiqueta}} --value  
{{valor_de_etiqueta}}
```

# az upgrade

Actualiza Azure CLI y sus extensiones.

Parte de **azure-cli** (también conocido como **az**).

Más información: <https://learn.microsoft.com/cli/azure/reference-index?view=azure-cli-latest#az-upgrade>.

- Actualiza Azure CLI:

```
az upgrade
```

- Actualiza Azure CLI y sus extensiones:

```
az upgrade --all
```

- Actualiza Azure CLI y sus extensiones sin solicitar confirmación:

```
az version --all --yes
```



# az vm

Gestiona máquinas virtuales en Azure.

Parte de **azure-cli** (también conocido como **az**).

Más información: <https://learn.microsoft.com/cli/azure/vm>.

- Muestra una tabla de Máquinas Virtuales disponibles:

```
az vm list --output table
```

- Crea una máquina virtual usando la imagen por defecto de Ubuntu y genera las claves SSH:

```
az vm create --resource-group {{rg}} --name {{nombre_vm}} --image {{UbuntuLTS}} --admin-user {{usuarioazure}} --generate-ssh-keys
```

- Detiene una Máquina Virtual:

```
az vm stop --resource-group {{rg}} --name {{nombre_de_la_máquina_virtual}}
```

- Desasigna una máquina virtual:

```
az vm deallocate --resource-group {{rg}} --name {{nombre_de_la_máquina_virtual}}
```

- Inicia una Máquina Virtual:

```
az vm start --resource-group {{rg}} --name {{nombre_de_la_máquina_virtual}}
```

- Reinicia una Máquina Virtual:

```
az vm restart --resource-group {{rg}} --name {{nombre_de_la_máquina_virtual}}
```

- Lista de imágenes de máquinas virtuales disponibles en Azure Marketplace:

```
az vm image list
```

# az webapp

Administra aplicaciones web alojadas en Azure Cloud Services.

Parte de **azure-cli** (también conocido como **az**)..

Más información: <https://learn.microsoft.com/cli/azure/webapp>.

- Lista los entornos de ejecución disponibles para una aplicación web:

```
az webapp list-runtimes --os-type {{windows|linux}}
```

- Crea una aplicación web:

```
az webapp up --name {{nombre}} --location {{ubicación}} --  
runtime {{entorno_de_ejecución}}
```

- Lista todas las aplicaciones web:

```
az webapp list
```

- Elimina una aplicación web específica:

```
az webapp delete --name {{nombre}} --resource-group  
{{grupo_de_recursos}}
```

# az

La herramienta de línea de comandos de Azure.

Algunos subcomandos como **login** tienen su propia documentación de uso.

Más información: <https://learn.microsoft.com/cli/azure>.

- Inicia sesión en Azure:

```
az login
```

- Administra la información de la suscripción de Azure:

```
az account
```

- Enumera todos los discos administrados de Azure:

```
az disk list
```

- Enumera todas las máquinas virtuales de Azure:

```
az vm list
```

- Administra los servicios de Kubernetes de Azure:

```
az aks
```

- Administra los recursos de red de Azure:

```
az network
```

# b2-tools

Accede fácilmente a todas las funciones de Backblaze B2 Cloud Storage.

Más información: <https://www.backblaze.com/docs/cloud-storage-command-line-tools>.

- Accede a tu cuenta:

```
b2 authorize_account {{clave_id}}
```

- Lista los buckets existentes en tu cuenta:

```
b2 list_buckets
```

- Crea un cubo, indica el nombre del cubo y el tipo de acceso (por ejemplo, allPublic o allPrivate):

```
b2 create_bucket {{nombre_cubo}} {{allPublic|allPrivate}}
```

- Sube un archivo. Elige un archivo, un bucket y una carpeta:

```
b2 upload_file {{nombre_cubo}} {{ruta/a/archivo}}  
{{nombre_carpeta}}
```

- Sube un directorio de origen a un destino de Backblaze B2 bucket:

```
b2 sync {{ruta/al/archivo_de_origen}} {{nombre_del_cubo}}
```

- Copia un archivo de un bucket a otro bucket:

```
b2 copy-file-by-id {{ruta/al/archivo_de_origen}}  
{{nombre_cubo_destino}} {{ruta/a/archivo/b2}}
```

- Muestra los archivos de tu bucket:

```
b2 ls {{nombre_bucket}}
```

- Elimina una "carpeta" o un conjunto de archivos que coincidan con un patrón:

```
b2 rm {{ruta/a/carpeta|patrón}}
```

# base32

Codifica o decodifica un archivo o la entrada estandar hacia/desde Base32, a la salida estandar.

Más información: <https://manned.org/base32>.

- Codifica un archivo:

```
base32 {{ruta/al/archivo}}
```

- Ajuste la salida codificada en un ancho específico (0 deshabilita el ajuste):

```
base32 {{-w|--wrap}} {{0|76|...}} {{ruta/al/archivo}}
```

- Decodifica un archivo:

```
base32 {{-d|--decode}} {{ruta/al/archivo}}
```

- Codifica stdin:

```
{{comando}} | base32
```

- Decodifica stdin:

```
{{comando}} | base32 {{-d|--decode}}
```

# base64

Codifica o decodifica un archivo o la entrada estandar hacia/desde Base64, a la salida estandar.

Más información: <https://manned.org/base64>.

- Codifica un archivo:

```
base64 {{ruta/al/archivo}}
```

- Ajuste la salida codificada en un ancho específico (0 deshabilita el ajuste):

```
base64 {{-w|--wrap}} {{0|76|...}} {{ruta/al/archivo}}
```

- Decodifica un archivo:

```
base64 {{-d|--decode}} {{ruta/al/archivo}}
```

- Codifica stdin:

```
{{comando}} | base64
```

- Decodifica stdin:

```
{{comando}} | base64 {{-d|--decode}}
```

# basename

Remueve nombres de directorios al inicio de una ruta.

Más información: <https://www.gnu.org/software/coreutils/basename>.

- Imprime el nombre de un fichero a partir de su ruta:

```
basename {{ruta/al/archivo}}
```

- Imprime el componente terminal de la ruta de un directorio:

```
basename {{ruta/al/directorio}}
```

- Imprime el nombre de un fichero sin un sufijo:

```
basename {{ruta/al/archivo}} {{sufijo}}
```

# bash

Bourne-Again SHell, un intérprete de línea de comandos compatible con **sh**.

Vea también: **zsh**; **histexpand**, para expansión de historial de comandos.

Más información: <https://www.gnu.org/software/bash/>.

- Inicia un intérprete de comandos interactivo:

```
bash
```

- Inicia el intérprete sin leer archivos de configuración:

```
bash --norc
```

- Ejecuta un comando:

```
bash -c "{{comando}}"
```

- Ejecuta comandos desde un archivo:

```
bash {{archivo.sh}}
```

- Ejecuta comandos desde un archivo, mostrando todos los comando ejecutados en la terminal:

```
bash -x {{archivo.sh}}
```

- Ejecuta comandos desde un archivo, deteniéndose en el primer error:

```
bash -e {{archivo.sh}}
```

- Ejecuta comandos desde `stdin` (entrada estándar):

```
{{echo "echo 'bash es ejecutado'"}} | bash
```

- Inicia el intérprete [r]estringido:

```
bash -r
```



# bat

Imprime y concatena archivos.

Un clon de **cat** con resaltado de sintaxis e integración con Git.

Más información: <https://github.com/sharkdp/bat>.

- Imprime los contenidos de un archivo a la salida estándar:

```
bat {{archivo}}
```

- Concatena varios archivos creando un nuevo archivo:

```
bat {{archivo1}} {{archivo2}} > {{archivo_final}}
```

- Añade múltiples archivos al final de un archivo objetivo:

```
bat {{archivo1}} {{archivo2}} >> {{archivo_final}}
```

- Numera las líneas del archivo:

```
bat --number {{archivo}}
```

- Muestra un archivo JSON con resaltado de sintaxis:

```
bat --language json {{archivo.json}}
```

- Muestra todos los lenguajes permitidos:

```
bat --list-languages
```

# bats

Bash Automated Testing System: un marco de pruebas compatible con TAP (<https://testanything.org/>) para Bash.

Más información: <https://bats-core.readthedocs.io/en/stable/usage.html>.

- Ejecuta un guión de prueba BATS y emite los resultados en el formato TAP (Test Anything Protocol):

```
bats --tap {{ruta/a/prueba.bats}}
```

- Cuenta los casos de prueba de un guión de prueba sin ejecutar ninguna prueba:

```
bats --count {{ruta/a/prueba.bats}}
```

- Ejecuta casos de prueba BATS en un directorio y sus subdirectorios (archivos con extensión .bats):

```
bats --recursive {{ruta/al/directorio}}
```

- Obtén resultados en un formato específico:

```
bats --formatter {{pretty|tap|junit}} {{ruta/a/prueba.bats}}
```

# bc

Un lenguaje de calculadora de precisión arbitraria.

Vea también: **dc**.

Más información: <https://manned.org/bc>.

- Inicia una sesión interactiva:

```
bc
```

- Inicia una sesión interactiva con la biblioteca matemática estándar activada:

```
bc --mathlib
```

- Calcula una expresión:

```
echo '{{5 / 3}}' | bc
```

- Ejecuta un script:

```
bc {{ruta/al/script.bc}}
```

- Calcula una expresión con la escala especificada:

```
echo 'scale = {{10}}; {{5 / 3}}' | bc
```

- Calcula una función seno/coseno/arctangente/logaritmo natural/exponencial utilizando `mathlib`:

```
echo '{{s|c|a|l|e}}({{1}})' | bc --mathlib
```

# bfs

Búsqueda exhaustiva de tus archivos.

Más información: <https://manned.org/bfs>.

- Busca archivos por extensión:

```
bfs {{ruta_raíz}} -name '{{*.ext}}'
```

- Busca archivos que coincidan con varios patrones de ruta/nombre:

```
bfs {{ruta_raíz}} -path '{{**/ruta/**/*.*ext}}' -or -name  
'{{*patrón*}}'
```

- Busca directorios que coincidan con un nombre dado, sin distinguir mayúsculas de minúsculas:

```
bfs {{ruta_raíz}} -type d -iname '{{*lib*}}'
```

- Busca archivos que coincidan con un patrón dado, excluyendo rutas específicas:

```
bfs {{ruta_raíz}} -name '{{*.py}}' -not -path '{{*/paquetes/  
*}}'
```

- Busca archivos que coincidan con un rango de tamaño dado, limitando la profundidad recursiva a "1":

```
bfs {{ruta_raíz}} -maxdepth 1 -size {{+500k}} -size {{-10M}}
```

- Ejecuta un comando para cada archivo (utiliza {} dentro del comando para acceder al nombre del archivo):

```
bfs {{ruta_root}} -name '{{*.ext}}' -exec {{wc -l}} {} \;
```

- Busca todos los archivos modificados hoy y pasa los resultados a un único comando como argumentos:

```
bfs {{ruta_raíz}} -daystart -mtime {{-1}} -exec {{tar -cvf  
archivo.tar}} {} \+
```

- Encuentra archivos vacíos (0 bytes) o directorios y los elimina de forma detallada:

```
bfs {{ruta_raíz}} -type {{f|d}} -empty -delete -print
```

# bitcoind

Daemon de Bitcoin Core.

Utiliza la configuración definida en **bitcoin.conf**.

Más información: <https://manned.org/bitcoind>.

- Inicia el daemon Bitcoin Core (en primer plano):

```
bitcoind
```

- Inicia el daemon Bitcoin Core en segundo plano (usa `bitcoin-cli stop` para detenerlo):

```
bitcoind -daemon
```

- Inicia el daemon Bitcoin Core en una red específica:

```
bitcoind -chain={{main|test|signet|regtest}}
```

- Inicia el daemon Bitcoin Core usando un archivo de configuración y directorio de datos específicos:

```
bitcoind -conf={{ruta/a/bitcoin.conf}} -datadir={{ruta/a/directorio}}
```

# bmptoppm

Este comando es reemplazado por **bmptopnm**.

Más información: <https://netpbm.sourceforge.net/doc/bmptoppm.html>.

- Ve documentación del comando actual:

```
tldr bmptopnm
```

# bpkg

Un gestor de paquetes para scripts Bash.

Más información: <https://github.com/bpkg/bpkg>.

- Actualiza el índice local:

```
bpkg update
```

- Instala un paquete globalmente:

```
bpkg install --global {{paquete}}
```

- Instala un paquete en un subdirectorio del directorio actual:

```
bpkg install {{paquete}}
```

- Instala una versión específica de un paquete de forma global:

```
bpkg install {{paquete}}@{{versión}} -g
```

- Muestra detalles sobre un paquete específico:

```
bpkg show {{paquete}}
```

- Ejecuta un comando, especificando opcionalmente sus argumentos:

```
bpkg run {{comando}} {{argumento1 argumento2 ...}}
```

# bpytop

Un monitor de recursos que muestra información sobre el CPU, la memoria, los discos, las redes y los procesos.

Una versión de **bashtop** en Python.

Más información: <https://github.com/aristocratos/bpytop>.

- Inicia bpytop:

```
bpytop
```

- Inicia en el modo mínimo sin los recuadros de memoria y redes:

```
bpytop -m
```

- Cambia a el modo mínimo:

```
m
```

- Busca procesos o programas en ejecución:

```
f
```

- Cambia los ajustes:

```
M
```

- Muestra la versión:

```
bpytop -v
```



# brew

Administrador de paquetes para macOS y Linux.

Más información: <https://docs.brew.sh/Manpage>.

- Instala la última versión estable de una fórmula (usar `--devel` para versiones de desarrollo):

```
brew install {{formula}}
```

- Lista todas las fórmulas y casks instaladas:

```
brew list
```

- Actualiza una fórmula o cask instalada (si no se indica ninguna, todas las fórmulas/casks se actualizan):

```
brew upgrade {{formula}}
```

- Trae la versión más reciente de Homebrew y todas sus fórmulas y casks desde el repositorio fuente de Homebrew:

```
brew update
```

- Muestra las fórmulas y casks que tienen una versión más reciente disponible:

```
brew outdated
```

- Busca fórmulas (por ej. paquetes) y casks (por ej. paquetes nativos) disponibles:

```
brew search {{texto}}
```

- Muestra la información de una fórmula o un cask (versión, ruta de instalación, dependencias, etc.):

```
brew info {{formula}}
```

- Revisa la instalación local de Homebrew en busca de problemas potenciales:

```
brew doctor
```

# bru

CLI para Bruno, un IDE de código abierto para explorar y probar APIs.

Más información: <https://docs.usebruno.com/bru-cli/overview>.

- Ejecuta todos los archivos de petición en el directorio actual:

```
bru run
```

- Ejecuta una única petición en el directorio actual especificando su nombre de archivo:

```
bru run {{archivo.bru}}
```

- Ejecuta peticiones utilizando un entorno:

```
bru run --env {{nombre_entorno}}
```

- Ejecuta peticiones utilizando un entorno con una variable:

```
bru run --env {{nombre_entorno}} --env-var  
{{nombre_variable}}={{valor_variable}}
```

- Ejecuta la solicitud y guarda los resultados en un archivo de salida:

```
bru run --output {{ruta/a/archivo_de_salida.json}}
```

- Muestra ayuda:

```
bru run --help
```

# btop

Un monitor de recursos que muestra información sobre la CPU, memoria, discos, red y procesos.

Versión en C++ de **bpytop**.

Más información: <https://github.com/aristocratos/btop>.

- Inicia btop:

```
btop
```

- Inicia btop con una configuración preestablecida:

```
btop --preset {{0..9}}
```

# bundler

Este comando es un alias de **bundle**.

Más información: <https://bundler.io/man/bundle.1.html>.

- Muestra la documentación del comando original:

```
tldr bundle
```

# calc

Una calculadora interactiva de precisión arbitraria en el terminal.

Más información: <https://github.com/lcn2/calc>.

- Inicia `calc` en modo interactivo:

```
calc
```

- Realiza un cálculo en modo no interactivo:

```
calc '{{85 * (36 / 4)}}'
```

- Realiza un cálculo sin ningún formato de salida:

```
calc -p '{{4/3 * pi() * 5^3}}'
```

- Realiza un cálculo y, a continuación, cambia a modo [i]nteractivo:

```
calc -i '{{sqrt(2)}}'
```

- Inicia `calc` en un [m]odo de permiso específico (de 0 a 7, por defecto 7):

```
calc -m {{modo}}
```

- Muestra una introducción a `calc`:

```
calc help intro
```

- Muestra una visión general de `calc`:

```
calc help overview
```

- Abre el manual de `calc`:

```
calc help
```

# cargo

Gestiona proyectos Rust y sus dependencias de módulos (crates).

Algunos subcomandos como **build** tienen su propia documentación de uso.

Más información: <https://doc.rust-lang.org/cargo>.

- Busca crates:

```
cargo search {{cadena_de_busqueda}}
```

- Instala un crate binario:

```
cargo install {{nombre_crate}}
```

- Lista los crates binarios instalados:

```
cargo install --list
```

- Crea un nuevo proyecto Rust binario o de biblioteca en el directorio especificado (o en el directorio de trabajo actual por defecto):

```
cargo init --{{bin|lib}} {{ruta/al/directorio}}
```

- Añade una dependencia a `Cargo.toml` en el directorio actual:

```
cargo add {{dependencia}}
```

- Construye el proyecto Rust en el directorio actual utilizando el perfil de lanzamiento:

```
cargo build --release
```

- Construye el proyecto Rust en el directorio actual utilizando el compilador nightly (requiere `rustup`):

```
cargo +nightly build
```

- Construye usando un número específico de hilos (por defecto es el número de CPUs lógicas):

```
cargo build --jobs {{numero_de_hilos}}
```

# cat

Imprime y concatena archivos.

Más información: <https://manned.org/cat.1posix>.

- Imprime el contenido de un fichero a `stdout`:

```
cat {{ruta/al/archivo}}
```

- Concatena varios archivos en un archivo de salida:

```
cat {{ruta/al/archivo1 ruta/al/archivo2 ...}} > {{ruta/al/archivo_salida}}
```

- Añade el contenido de varios archivos a un archivo de salida:

```
cat {{ruta/al/archivo1 ruta/al/archivo2 ...}} >> {{ruta/al/archivo_salida}}
```

- Copia el contenido de un archivo en un archivo de salida sin almacenamiento en el búfer:

```
cat -u {{/dev/tty12}} > {{/dev/tty13}}
```

- Copia `stdin` en un archivo:

```
cat - > {{ruta/al/archivo}}
```

# ccache

Caché del compilador C/C++.

Nota: los paquetes suelen proporcionar enlaces simbólicos para los compiladores en `/usr/lib/ccache/bin`. Anteponga este directorio en `$PATH` para utilizar automáticamente **ccache** con los compiladores.

Más información: <https://ccache.dev/manual/latest.html>.

- Muestra las e[s]tadísticas de la caché actual:

```
ccache --show-stats
```

- Borra toda la caché:

```
ccache --clear
```

- Restablece ([z]ero) las estadísticas (pero no la propia caché):

```
ccache --zero-stats
```

- Compila código C y almacena la salida compilada en la caché (para usar **ccache** en todas las invocaciones de **gcc**, lea la nota anterior):

```
ccache gcc {{ruta/al/archivo.c}}
```



# cd

Cambia el directorio de trabajo actual.

Más información: <https://manned.org/cd>.

- Accede al directorio especificado:

```
cd {{ruta/al/directorio}}
```

- Cambia al directorio padre:

```
cd ..
```

- Accede al directorio raíz del usuario actual:

```
cd
```

- Accede al directorio personal del usuario especificado:

```
cd ~{{nombredeusuario}}
```

- Cambia al directorio elegido anteriormente:

```
cd -
```

- Cambia al directorio raíz:

```
cd /
```

# chatgpt

Shell script para usar ChatGPT de OpenAI y DALL-E desde la terminal.

Más información: <https://github.com/0xacx/chatGPT-shell-cli>.

- Comienza en modo chat:

```
chatgpt
```

- Da un [p]rompt para responder:

```
chatgpt --prompt "{{¿Cuál es la expresión regular para  
emparejar una dirección de correo electrónico?}}"
```

- Inicia en modo chat utilizando un [m]odelo específico (por defecto es gpt-3.5-turbo):

```
chatgpt --model {{gpt-4}}
```

- Inicia en modo chat con un prompt [i]nicial:

```
chatgpt --init-prompt "{{Tú eres Rick, de Rick y Morty.  
Responde a las preguntas usando su amaneramiento e incluye  
chistes insultantes.}}"
```

- Envía el resultado de un comando a ChatGPT como un prompt:

```
echo "{{¿Cómo ver los procesos en ejecución en Ubuntu?}}" |  
chatgpt
```

- Genera una imagen utilizando DALL-E:

```
chatgpt --prompt "{{image: Un gato blanco}}"
```

# checkov

Checkov es una herramienta de análisis estático de código para Infraestructura como Código (IaC).

También es una herramienta de análisis de composición de software (SCA) para imágenes y paquetes de código abierto.

Más información: <https://www.checkov.io/1.Welcome/Quick%20Start.html>.

- Analiza un directorio que contenga IaC (Terraform, Cloudformation, ARM, Ansible, Bicep, Dockerfile, etc):

```
checkov --directory {{ruta/al/directorio}}
```

- Escanea un archivo IaC, omitiendo bloques de código en la salida:

```
checkov --compact --file {{ruta/al/archivo}}
```

- Lista todas las comprobaciones de todos los tipos de IaC:

```
checkov --list
```

# chmod

Cambia los permisos de acceso de un archivo o directorio.

Más información: <https://www.gnu.org/software/coreutils/chmod>.

- Otorga al [u]suario que es propietario del archivo permiso para [x] ejecutarlo:

```
chmod u+x {{ruta/al/archivo}}
```

- Otorga al [u]suario derechos para leer (r) y escribir (w) un archivo o directorio:

```
chmod u+rw {{ruta/al/archivo_o_directorio}}
```

- Elimina los derechos de ejecución del [g]ruppo:

```
chmod g-x {{ruta/al/archivo}}
```

- Otorga a todos los usuarios (a) derechos para leer y ejecutar:

```
chmod a+rx {{ruta/al/archivo}}
```

- Otorga a [o]tros (que no están en el grupo del propietario) los mismos derechos que los del [g]ruppo:

```
chmod o=g {{ruta/al/archivo}}
```

- Quita todos los derechos a [o]tros:

```
chmod o= {{ruta/al/archivo}}
```

- Otorga al [g]ruppo y a [o]tros el derecho para escribir (w) un directorio y su contenido:

```
chmod -R g+w,o+w {{ruta/al/directorio}}
```

- Concede de forma recursiva [a] todos los usuarios permisos de lectu[r]a a los archivos y permisos de e[X]ecución a los subdirectorios dentro de un directorio:

```
chmod -R a+rX {{ruta/al/directorio}}
```

# chown

Cambia la propiedad de usuario y grupo sobre archivos y directorios.

Más información: <https://www.gnu.org/software/coreutils/chown>.

- Cambia el usuario propietario de un archivo/directorio:

```
chown {{usuario}} {{ruta/hacia/archivo_o_directorio}}
```

- Cambia el usuario y grupo propietario de un archivo/directorio:

```
chown {{usuario}}:{{grupo}} {{ruta/hacia/  
archivo_o_directorio}}
```

- Cambia de forma recursiva el propietario sobre un directorio y su contenido:

```
chown -R {{usuario}} {{ruta/hacia/directorio}}
```

- Cambia el propietario de un enlace simbólico:

```
chown -h {{usuario}} {{ruta/hacia/enlace_simbolico}}
```

- Copia la información de propiedad del archivo/directorio de referencia a otro:

```
chown --reference={{ruta/hacia/  
archivo_o_directorio_de_referencia}} {{ruta/hacia/  
archivo_o_directorio}}
```

# chsh

Cambia el intérprete de comandos de inicio de sesión.

Más información: <https://manned.org/chsh>.

- Cambia interactivamente el intérprete del usuario actual:

```
chsh
```

- Cambia el intérprete del usuario actual por otro específico:

```
chsh -s {{ruta/al/intérprete}}
```

- Cambia el intérprete de otro usuario:

```
chsh -s {{ruta/al/intérprete}} {{nombre_de_usuario}}
```

- [l]ista los intérpretes disponibles:

```
chsh -l
```

# cjxl

Comprime imágenes a JPEG XL.

Las extensiones de entrada aceptadas son PNG, APNG, GIF, JPEG, EXR, PPM, PFM, PAM, PGX y JXL.

Más información: <https://github.com/libjxl/libjxl>.

- Convierte una imagen a JPEG XL:

```
cjxl {{ruta/a/imagen.ext}} {{ruta/a/salida.jxl}}
```

- Ajusta la calidad a sin pérdidas y maximiza la compresión de la imagen resultante:

```
cjxl --distance 0 --effort 9 {{ruta/a/imagen.ext}} {{ruta/a/salida.jxl}}
```

- Muestra una página de ayuda muy detallada:

```
cjxl --help --verbose --verbose --verbose --verbose
```

# ClamAV

Programa antivirus de código abierto.

ClamAV no es un comando, sino un conjunto de comandos.

Más información: <https://www.clamav.net>.

- Muestra la página tldr para escanear archivos usando el daemon `clamd`:

```
tldr clamdscan
```

- Muestra la página tldr para escanear archivos sin que se ejecute el daemon `clamd`:

```
tldr clamscan
```

- Muestra la página tldr para actualizar las definiciones de virus:

```
tldr freshclam
```



# clang-cpp

Este comando es un alias de **clang++**.

- Muestra la documentación del comando original:

```
tldr clang++
```

# clear

Limpia la pantalla de la terminal.

Más información: <https://manned.org/clear>.

- Limpia la pantalla de la terminal (equivale a presionar Control-L en la interfaz de línea de comandos Bash):

```
clear
```

- Limpia la pantalla pero mantiene el buffer de desplazamiento:

```
clear -x
```

- Indica el tipo de terminal a limpiar (por defecto se utiliza el valor de la variable de entorno `TERM`):

```
clear -T {{tipo_de_terminal}}
```

- Muestra la versión de `ncurses` utilizada por `clear`:

```
clear -V
```

# clojure

Este comando es un alias de **clj**.

- Muestra la documentación del comando original:

```
tldr clj
```

# code

Editor de código extensible y multiplataforma.

Más información: <https://github.com/microsoft/vscode>.

- Inicia Visual Studio Code:

```
code
```

- Abre archivos o directorios específicos:

```
code {{ruta/al/archivo_o_directorio1 ruta/al/
archivo_o_directorio2 ...}}
```

- Compara dos archivos específicos:

```
code --diff {{ruta/al/archivo1}} {{ruta/al/archivo2}}
```

- Abre archivos o directorios específicos en una nueva ventana:

```
code --new-window {{ruta/al/archivo_o_directorio1 ruta/al/
archivo_o_directorio2 ...}}
```

- Instala/desinstala una extensión específica:

```
code --{{install|uninstall}}-extension {{editor.extension}}
```

- Imprime las extensiones instaladas:

```
code --list-extensions
```

- Imprime las extensiones instaladas con su versión:

```
code --list-extensions --show-versions
```

- Inicia el editor como súper usuario (root) mientras que almacena los datos del usuario en un directorio específico:

```
sudo code --user-data-dir {{ruta/al/directorio}}
```

# codecrafters

Practica escribiendo software complejo.

Más información: <https://codecrafters.io/>.

- Ejecuta pruebas sin confirmar cambios:

```
codecrafters test
```

- Ejecuta pruebas para todas las etapas anteriores y la etapa actual sin confirmar los cambios:

```
codecrafters test --previous
```

- Confirma los cambios y los envía, para pasar a la siguiente fase:

```
codecrafters submit
```

# cola

Este comando es un alias de **git-cola**.

- Muestra la documentación del comando original:

```
tldr git-cola
```

# colima

Contenedores en tiempo de ejecución para macOS y Linux con una configuración mínima.

Más información: <https://github.com/abiosoft/colima>.

- Inicia el daemon en segundo plano:

```
colima start
```

- Crea un archivo de configuración y lo usa:

```
colima start --edit
```

- Inicia y configura containerd (instala `nerdctl` para usar containerd a través de `nerdctl`):

```
colima start --runtime containerd
```

- Inicia con Kubernetes (se requiere `kubectl`):

```
colima start --kubernetes
```

- Personaliza el recuento de CPU, memoria RAM y espacio en disco (en GiB):

```
colima start --cpu {{número}} --memory {{memoria}} --disk {{espacio_de_almacenamiento}}
```

- Usa Docker a través de Colima (se requiere Docker):

```
colima start
```

- Lista contenedores con su información y estado:

```
colima list
```

- Muestra estado de tiempo de ejecución:

```
colima status
```

# cp

Copia archivos y directorios.

Más información: <https://www.gnu.org/software/coreutils/cp>.

- Copia un archivo a otra ruta:

```
cp {{ruta/hacia/archivo_original.ext}} {{ruta/hacia/archivo_copia.ext}}
```

- Copia un archivo a un directorio, manteniendo el nombre del archivo:

```
cp {{ruta/hacia/archivo_original.ext}} {{ruta/hacia/directorio_destino}}
```

- Copia de forma recursiva un directorio y su contenido a otra ruta (si la ruta de destino existe, el directorio se copiará dentro):

```
cp -R {{ruta/hacia/directorio_original}} {{ruta/hacia/directorio_copia}}
```

- Copia de forma recursiva y verbosa un directorio (muestra un listado de los archivos copiados):

```
cp -vR {{ruta/hacia/directorio_original}} {{ruta/hacia/directorio_copia}}
```

- Copia archivos de texto a otra ruta de forma interactiva (pregunta al usuario antes de sobrescribir):

```
cp -i {{*.txt}} {{ruta/hacia/directorio_destino}}
```

- Copia enlaces simbólicos sin mantener la referencia al original:

```
cp -L {{enlace}} {{ruta/hacia/directorio_destino}}
```



# cpdf

Interfaz de línea de comandos para manipular documentos PDF existentes de diferentes maneras.

Más información: <https://www.coherentpdf.com/cpdfmanual/cpdfmanual.html>.

- Selecciona las páginas 1, 2, 3 y 6 del documento fuente y las agrega en el documento objetivo:

```
cpdf {{ruta/al/documento_fuente.pdf}} {{1-3,6}} -o {{ruta/al/documento_objetivo.pdf}}
```

- Fusiona dos documentos en uno nuevo:

```
cpdf -merge {{ruta/al/documento_fuente_uno.pdf}} {{ruta/al/documento_fuente_dos.pdf}} -o {{ruta/al/documento_objetivo.pdf}}
```

- Muestra los marcadores del documento:

```
cpdf -list-bookmarks {{ruta/al/documento.pdf}}
```

- Divide un documento en trozos de diez páginas, escribiendo fragmento001.pdf, fragmento002.pdf, etc:

```
cpdf -split {{ruta/al/documento.pdf}} -o {{ruta/al/fragmento%  
%.pdf}} -chunk {{10}}
```

- Encripta un documento utilizando encriptado 128bit y establece fred como la contraseña del propietario y joe como la contraseña de usuario:

```
cpdf -encrypt {{128bit}} {{fred}} {{joe}} {{ruta/al/documento_fuente.pdf}} -o {{ruta/al/documento_encriptado.pdf}}
```

- Desencripta un documento utilizando la contraseña del propietario (fred):

```
cpdf -decrypt {{ruta/al/documento_encriptado.pdf}}  
owner={{fred}} -o {{ruta/al/documento_desencriptado.pdf}}
```

- Muestra las anotaciones de un documento:

```
cpdf -list-annotations {{ruta/al/documento.pdf}}
```

- Crea un nuevo documento, con metadatos, a partir de uno que ya existe:

```
cpdf -set-metadata {{ruta/de/los/metadatos.xml}} {{ruta/al/documento_fuente.pdf}} -o {{ruta/al/documento_objetivo.pdf}}
```

# crackle

Crackea y descifra el cifrado Bluetooth Low Energy (BLE).

Más información: <https://github.com/mikeryan/crackle>.

- Comprueba si las comunicaciones BLE grabadas contienen los paquetes necesarios para recuperar claves temporales (TKs):

```
crackle -i {{ruta/a/entrada.pcap}}
```

- Utiliza la fuerza bruta para recuperar la TK de los eventos de emparejamiento registrados y la utiliza para descifrar todas las comunicaciones posteriores:

```
crackle -i {{ruta/a/entrada.pcap}} -o {{ruta/a/desencriptado.pcap}}
```

- Utiliza la clave a largo plazo (LTK) especificada para descifrar la comunicación grabada:

```
crackle -i {{ruta/a/entrada.pcap}} -o {{ruta/a/descifrar.pcap}} -l {{81b06facd90fe7a6e9bbd9cee59736a7}}
```

# cron

Este comando es un alias de **crontab**.

- Muestra la documentación del comando original:

```
tldr crontab
```

# ctags

Genera un archivo de índice (o etiqueta) de objetos de lenguaje que se encuentran en los archivos de código fuente de muchos lenguajes de programación populares.

Más información: <https://ctags.io/>.

- Genera etiquetas para un solo archivo y las envía a un archivo llamado "tags" en el directorio actual, sobrescribiendo el archivo si existe:

```
ctags {{ruta/al/archivo}}
```

- Genera etiquetas para todos los archivos en el directorio actual y las envía a un archivo específico, sobrescribiendo el archivo si existe:

```
ctags -f {{ruta/al/archivo}} *
```

- Genera etiquetas para todos los archivos en el directorio actual y todos los subdirectorios:

```
ctags --recurse
```

- Genera etiquetas para un solo archivo y las envía con el número de línea inicial y el número de línea final en formato JSON:

```
ctags --fields=+ne --output-format=json {{ruta/al/archivo}}
```

# cut

Corta campos de **stdin** o archivos.

Más información: <https://www.gnu.org/software/coreutils/cut>.

- Imprime un rango específico de caracteres/campos de cada línea:

```
{{comando}} | cut --{{characters|field}} {{1|1,10|1-10|  
1-|-10}}
```

- Imprime un rango de campos de cada línea con un [d]elimitador específico:

```
{{comando}} | cut --delimiter "{{{,}}}" --fields {{1}}
```

- Imprime un rango de [c]aracteres de cada línea del archivo específico:

```
cut --characters {{1}} {{ruta/al/archivo}}
```

- Imprime [c]ampos específicos de líneas terminadas en **NUL** (por ejemplo, como en `find . -print0`) en lugar de nuevas líneas:

```
{{comando}} | cut --zero-terminated --fields {{1}}
```

# cypher-shell

Abre una sesión interactiva y ejecuta consultas Cypher contra una instancia Neo4j.

Vea también: **neo4j-admin**, **mysql**.

Más información: <https://neo4j.com/docs/operations-manual/current/tools/cypher-shell/>.

- Conéctate a una instancia local en el puerto por defecto (neo4j://localhost:7687):  
`cypher-shell`
- Conéctate a una instancia remota:  
`cypher-shell --address neo4j://{{host}}:{{puerto}}`
- Conéctate y proporciona credenciales de seguridad:  
`cypher-shell --username {{nombre_de_usuario}} --password {{contraseña}}`
- Conéctate a una base de datos específica:  
`cypher-shell --database {{nombre_base_de_datos}}`
- Ejecuta sentencias Cypher en un archivo y lo cierra:  
`cypher-shell --file {{ruta/al/archivo.cypher}}`
- Activa el registro en un archivo:  
`cypher-shell --log {{ruta/al/archivo.log}}`
- Muestra ayuda:  
`cypher-shell --help`

# d2

Un lenguaje moderno de scripting de diagramas que convierte texto en diagramas.

Nota: el archivo de salida admite formatos de archivo SVG y PNG.

Más información: <https://d2lang.com/tour/man>.

- Compila y renderiza un archivo fuente D2 en un archivo de salida:

```
d2 {{ruta/al/archivo_de_entrada.d2}} {{ruta/al/
archivo_de_salida.ext}}
```

- Ve en directo los cambios realizados en un archivo fuente D2 en el navegador web predeterminado:

```
d2 --watch {{ruta/al/archivo_de_entrada.d2}} {{ruta/al/
archivo_de_salida.ext}}
```

- Formatea un archivo fuente D2:

```
d2 fmt {{ruta/al/archivo_de_entrada.d2}}
```

- Lista los temas disponibles:

```
d2 themes
```

- Usa un [t]ema diferente para el archivo de salida (primero enumera los temas disponibles para obtener el `theme_id` deseado):

```
d2 --theme {{identificador_tema}} {{ruta/al/
archivo_de_entrada.d2}} {{ruta/al/archivo_de_salida.ext}}
```

- Haz que los diagramas renderizados parezcan bocetos hechos a mano:

```
d2 --sketch true {{ruta/al/archivo_de_entrada.d2}} {{ruta/al/
archivo_de_salida.ext}}
```

# deno

Un entorno de ejecución seguro para JavaScript y TypeScript.

Más información: <https://deno.land>.

- Ejecuta un archivo JavaScript o TypeScript:

```
deno run {{ruta/al/archivo.ts}}
```

- Inicia un REPL (intérprete de comandos interactivo):

```
deno
```

- Ejecuta un archivo con acceso a la red habilitado:

```
deno run --allow-net {{ruta/al/archivo.ts}}
```

- Ejecuta un archivo desde una URL:

```
deno run {{https://deno.land/std/examples/welcome.ts}}
```

- Instala un archivo de secuencia de comandos ejecutable desde una URL:

```
deno install {{https://deno.land/std/examples/colors.ts}}
```



# devenv

Entornos de desarrollo rápidos, declarativos, reproducibles y componibles utilizando Nix.

Más información: <https://devenv.sh>.

- Inicializa el entorno:

```
devenv init
```

- Entra en el entorno de desarrollo con hermeticidad relajada (estado de ser hermético):

```
devenv shell --impure
```

- Obtén información detallada sobre el entorno actual:

```
devenv info --verbose
```

- Inicia procesos con devenv:

```
devenv up --config /{{archivo}}/{{ruta}}/
```

- Limpia las variables de entorno y vuelve a entrar en el intérprete de comandos en el modo sin conexión:

```
devenv --clean --offline
```

- Borra las generaciones anteriores del intérprete de comandos:

```
devenv gc
```

# df

Entrega información general del uso de espacio en disco del sistema de archivos.

Más información: <https://manned.org/df.1posix>.

- Muestra todos los sistemas de archivos y sus usos de disco:

```
df
```

- Muestra todos los sistemas de archivos y sus usos de disco en formato legible para humanos:

```
df -h
```

- Muestra el sistema de archivos que contiene determinado archivo o directorio y su uso de disco:

```
df {{ruta/al/archivo_o_directorio}}
```

- Muestra estadísticas sobre el número de inodos libres:

```
df -i
```

- Muestra sistemas de archivos excluyendo los tipos especificados:

```
df -x {{squashfs}} -x {{tmpfs}}
```

# diff

Compara archivos y directorios.

Más información: <https://manned.org/diff>.

- Compara archivos (lista los cambios para convertir `archivo_viejo` en `archivo_nuevo`):

```
diff {{archivo_viejo}} {{archivo_nuevo}}
```

- Compara archivos, ignorando los espacios en blanco:

```
diff {{-w|--ignore-all-space}} {{archivo_viejo}}  
{{archivo_nuevo}}
```

- Compara archivos, mostrando las diferencias lado a lado:

```
diff {{-y|--side-by-side}} {{archivo_viejo}}  
{{archivo_nuevo}}
```

- Compara archivos, mostrando las diferencias en formato unificado (como el que usa `git diff`):

```
diff {{-u|--unified}} {{archivo_viejo}} {{archivo_nuevo}}
```

- Compara directorios de forma recursiva (muestra los nombres de los archivos/directorios que difieran y los cambios realizados en los archivos):

```
diff {{-r|--recursive}} {{directorio_viejo}}  
{{directorio_nuevo}}
```

- Compara directorios, mostrando solo los nombres de los archivos que difieren:

```
diff {{-r|--recursive}} {{-q|--brief}} {{directorio_viejo}}  
{{directorio_nuevo}}
```

- Crea un archivo de revisión para Git a partir de las diferencias entre dos archivos de texto, tratando los archivos inexistentes como vacíos:

```
diff {{-a|--text}} {{-u|--unified}} {{-N|--new-file}}  
{{archivo_viejo}} {{archivo_nuevo}} > {{diff.patch}}
```

- Compara archivos, mostrando la salida en color y se esfuerza por encontrar el conjunto más pequeño de cambios:

```
diff {{-d|--minimal}} --color=always {{archivo_viejo}}  
{{archivo_nuevo}}
```

# diff

Compara archivos o directorios en base de la sintaxis del lenguaje de programación.

Vea también: **delta**, **diff**.

Más información: <https://difftastic.wilfred.me.uk/introduction.html>.

- Compara dos archivos o directorios:

```
diff {{ruta/al/archivo_o_directorio1}} {{ruta/al/archivo_o_directorio2}}
```

- Informa únicamente diferencias entre los archivos:

```
diff --check-only {{ruta/al/archivo1}} {{ruta/al/archivo2}}
```

- Especifica el modo de visualización (por defecto es *side-by-side*):

```
diff --display {{side-by-side|side-by-side-show-both|inline|json}} {{ruta/al/archivo1}} {{ruta/al/archivo2}}
```

- Ignora comentarios al comparar:

```
diff --ignore-comments {{ruta/al/archivo1}} {{ruta/al/archivo2}}
```

- Activa o desactiva el resaltado sintáctico del código fuente (por defecto está activado):

```
diff --syntax-highlight {{on|off}} {{ruta/al/archivo1}} {{ruta/al/archivo2}}
```

- Silenciosamente omite los archivos que no hayan cambiado:

```
diff --skip-unchanged {{ruta/al/archivo_o_directorio1}} {{ruta/al/archivo_o_directorio2}}
```

- Lista todos los lenguajes de programación soportados por la herramienta, junto con sus extensiones:

```
diff --list-languages
```

# dig

Utilidad de consulta para DNS.

Más información: <https://manned.org/dig>.

- Consulta la(s) IP(s) asociadas a un nombre de equipo (registros A):

```
dig +short {{example.com}}
```

- Obtén una respuesta detallada para un dominio determinado (registros A):

```
dig +noall +answer {{example.com}}
```

- Consulta un tipo de registro DNS específico asociado a un dominio determinado:

```
dig +short {{example.com}} {{A|MX|TXT|CNAME|NS}}
```

- Especifica un servidor DNS alternativo a consultar:

```
dig @{{8.8.8.8}} {{example.com}}
```

- Realiza una búsqueda DNS inversa para una dirección IP (registro PTR):

```
dig -x {{8.8.8.8}}
```

- Encuentra servidores de nombre autoritativos para la zona y muestra registros SOA:

```
dig +nssearch {{example.com}}
```

- Realiza consultas iterativas y muestra el trazado de ruta completo para resolver un dominio:

```
dig +trace {{example.com}}
```

# djxl

Descomprime imágenes JPEG XL.

Las extensiones de salida aceptadas son PNG, APNG, JPEG, EXR, PGM, PPM, PNM, PFM, PAM, EXIF, XMP y JUMBF.

Más información: <https://github.com/libjxl/libjxl>.

- Descomprime una imagen JPEG XL a otro formato:

```
djxl {{ruta/a/imagen.jxl}} {{ruta/a/salida.ext}}
```

- Muestra una página de ayuda muy detallada:

```
djxl --help --verbose --verbose --verbose --verbose
```

# dnsx

Un equipo de herramientas de DNS rápido y multipropósito para ejecutar múltiples consultas DNS.

Nota: la entrada a **dnsx** necesita ser pasada a través de **stdin** (tubería |) en algunos casos.

Ver también: **dig**, **dog**, **dnstracer**.

Más información: <https://github.com/projectdiscovery/dnsx>.

- Consulta el registro A de un subdominio y muestra la [re]spuesta recibida:

```
echo {{ejemplo.com}} | dnsx -a -re
```

- Consulta todos los registros DNS (A, AAAA, CNAME, NS, TXT, SRV, PTR, MX, SOA, AXFR y CAA):

```
dnsx -recon -re <<< {{ejemplo.com}}
```

- Consulta un tipo específico de registro DNS:

```
echo {{ejemplo.com}} | dnsx -re -{{a|aaaa|cname|ns|txt|srv|ptr|mx|soa|any|axfr|caa}}
```

- Muestra s[o]lo la [r]espuesta (no muestra el dominio o subdominio consultado):

```
echo {{ejemplo.com}} | dnsx -ro
```

- Muestra la respuesta sin procesar una consulta, especificando los solucionado[r]es a utilizar y el número de intentos en caso de haber errores:

```
echo {{ejemplo.com}} | dnsx -{{debug|raw}} -resolver {{1.1.1.1,8.8.8.8,...}} -retry {{número}}
```

- Aplica fuerza bruta a registros DNS utilizando un marcador de posición:

```
dnsx -domain {{FUZZ.ejemplo.com}} -wordlist {{ruta/a/ lista_de_palabras.txt}} -re
```

- Aplica fuerza bruta a registros DNS a partir de una lista de [d]ominios y listas de palabras, adjuntando la salida a un archivo sin códigos de [c]olor:



```
dnsx -domain {{ruta/a/ dominio.txt}} -wordlist {{ruta/a/
lista_de_palabras.txt}} -re -output {{ruta/a/salida.txt}} -
no-color
```

- Extrae registros **CNAME** desde una lista de subdominios, con una velocidad [l]ímite de consultas DNS por segundo:

```
subfinder -silent -d {{ejemplo.com}} | dnsx -cname -re -rl
{{número}}
```

# docker compose

Ejecuta y gestiona aplicaciones Docker multicontenedor.

Más información: <https://docs.docker.com/reference/cli/docker/compose/>.

- Lista todos los contenedores en ejecución:

```
docker compose ps
```

- Crea e inicia todos los contenedores en segundo plano usando un archivo `docker-compose.yml` desde el directorio actual:

```
docker compose up --detach
```

- Inicia todos los contenedores, y se reconstruye si es necesario:

```
docker compose up --build
```

- Inicia todos los contenedores especificando un nombre de proyecto y utilizando un archivo de composición alternativo:

```
docker compose -p {{nombre_proyecto}} --file {{ruta/al/archivo}} up
```

- Detiene todos los contenedores en ejecución:

```
docker compose stop
```

- Detiene y elimina todos los contenedores, redes, imágenes y volúmenes:

```
docker compose down --rmi all --volumes
```

- Sigue los registros de todos los contenedores:

```
docker compose logs --follow
```

- Sigue los registros de un contenedor específico:

```
docker compose logs --follow {{nombre_del_contenedor}}
```

# docker

Administra contenedores e imágenes de Docker.

Algunos subcomandos, como **run**, tienen su propia documentación de uso.

Más información: <https://docs.docker.com/reference/cli/docker/>.

- Lista todos los contenedores de Docker (en ejecución y detenidos):

```
docker ps --all
```

- Inicia un contenedor desde una imagen con un nombre personalizado:

```
docker run --name {{nombre_de_contenedor}} {{imagen}}
```

- Inicia o detiene un contenedor existente:

```
docker {{start|stop}} {{nombre_de_contenedor}}
```

- Descarga una imagen desde un registro de Docker:

```
docker pull {{imagen}}
```

- Muestra la lista de imágenes descargadas:

```
docker images
```

- Inicia una línea de comandos dentro de un contenedor en ejecución:

```
docker exec -it {{nombre_de_contenedor}} {{sh}}
```

- Elimina un contenedor detenido:

```
docker rm {{nombre_de_contenedor}}
```

- Obtén y sigue los registros de un contenedor:

```
docker logs -f {{nombre_de_contenedor}}
```

# doggo

Cliente DNS para Humanos.

Escrito en Golang.

Más información: <https://github.com/mr-karan/doggo>.

- Realiza una simple búsqueda DNS:

```
doggo {{example.com}}
```

- Consulta registros MX usando un servidor de nombres específico:

```
doggo MX {{codeberg.org}} @{{1.1.1.2}}
```

- Utiliza DNS sobre HTTPS:

```
doggo {{example.com}} @{{https://dns.quad9.net/dns-query}}
```

- Salida en formato JSON:

```
doggo {{example.com}} --json | jq  
'{{.responses[0].answers[0].address}}'
```

- Realiza una búsqueda DNS inversa:

```
doggo --reverse {{8.8.4.4}} --short
```

# dolt gc

Busca en el repositorio los datos que ya no se referencian ni necesitan.

Más información: <https://docs.dolthub.com/cli-reference/cli#dolt-gc>.

- Limpia datos no referenciados del repositorio:

```
dolt gc
```

- Inicia un proceso de recolección de basura más rápido pero menos exhaustivo:

```
dolt gc --shallow
```

# doppler

Gestiona variables de entorno a través de diferentes entornos usando Doppler.

Algunos subcomandos como **run** y **secrets** tienen su propia documentación de uso.

Más información: <https://docs.doppler.com/docs/cli>.

- Configura Doppler CLI en el directorio actual:

```
doppler setup
```

- Configura el proyecto Doppler y la configuración en el directorio actual:

```
doppler setup
```

- Ejecuta un comando con secretos inyectados en el entorno:

```
doppler run --command {{comando}}
```

- Visualiza la lista de proyectos:

```
doppler projects
```

- Visualiza los secretos del proyecto actual:

```
doppler secrets
```

- Abre el panel de control de doppler en el navegador:

```
doppler open
```

# dotenvx

Un **dotenv** mejor, del creador de **dotenv**.

Más información: <https://dotenvx.com/docs>.

- Ejecuta un comando con variables de entorno desde un archivo `.env`:

```
dotenvx run -- {{comando}}
```

- Ejecuta un comando con variables de entorno desde un archivo `.env` específico:

```
dotenvx run -f {{ruta/al/archivo.env}} -- {{command}}
```

- Establece una variable de entorno con cifrado:

```
dotenvx set {{clave}} {{valor}}
```

- Establece una variable de entorno sin encriptación:

```
dotenvx set {{clave}} {{valor}} --plain
```

- Devuelve las variables de entorno definidas en un archivo `.env`:

```
dotenvx get
```

- Devuelve el valor de una variable de entorno definida en un archivo `.env`:

```
dotenvx get {{clave}}
```

- Devuelve todas las variables de entorno de los archivos `.env` y OS:

```
dotenvx get --all
```

# dotnet build

Compila una aplicación .NET y sus dependencias.

Más información: <https://learn.microsoft.com/dotnet/core/tools/dotnet-build>.

- Compila el proyecto o solución en el directorio actual:

```
dotnet build
```

- Compila un proyecto o solución .NET en el modo de depuración:

```
dotnet build {{ruta/al/proyecto_o_solución}}
```

- Compila en modo de lanzamiento:

```
dotnet build --configuration {{Release}}
```

- Compila sin restaurar las dependencias:

```
dotnet build --no-restore
```

- Compila con un nivel específico de verbosidad:

```
dotnet build --verbosity {{quiet|minimal|normal|detailed|  
diagnostic}}
```

- Compila para un tiempo de ejecución específico:

```
dotnet build --runtime  
{{identificador_del_tiempo_de_ejecución}}
```

- Especifica el directorio de salida:

```
dotnet build --output {{ruta/al/directorio}}
```



# dotnet publish

Publica una aplicación .NET y sus dependencias en una carpeta para la implementación en un sistema de hospedaje.

Más información: <https://learn.microsoft.com/dotnet/core/tools/dotnet-publish>.

- Compila un proyecto .NET en modo de lanzamiento:

```
dotnet publish --configuration Release {{ruta/al/  
archivo_del_proyecto}}
```

- Publica el entorno de ejecución de .NET Core con la aplicación para un entorno de ejecución específico:

```
dotnet publish --self-contained true --runtime  
{{identificador_del_entorno_en_tiempo_de_ejecución}} {{ruta/  
al/archivo_del_proyecto}}
```

- Empaqueta la aplicación en un archivo ejecutable único de una plataforma específica:

```
dotnet publish --runtime  
{{identificador_del_entorno_en_tiempo_de_ejecución}} -  
p:PublishSingleFile=true {{ruta/al/archivo_del_proyecto}}
```

- Recorta las bibliotecas no usadas para reducir el tamaño de la aplicación:

```
dotnet publish --self-contained true --runtime  
{{identificador_del_entorno_de_tiempo_de_ejecución}} -  
p:PublishTrimmed=true {{ruta/al/archivo_del_proyecto}}
```

- Compila un proyecto .NET sin restaurar las dependencias:

```
dotnet publish --no-restore {{ruta/al/archivo_del_proyecto}}
```

- Especifica el directorio de salida:

```
dotnet publish --output {{ruta/al/directorio}} {{ruta/al/  
archivo_del_proyecto}}
```

# dotnet restore

Restaura las dependencias y herramientas de un proyecto .NET.

Más información: <https://learn.microsoft.com/dotnet/core/tools/dotnet-restore>.

- Restaura dependencias para un proyecto o solución .NET en el directorio actual:

```
dotnet restore
```

- Restaura dependencias para un proyecto o solución .NET en una ubicación específica:

```
dotnet restore {{ruta/al/proyecto_o_solución}}
```

- Restaura dependencias sin almacenar las solicitudes HTTP en caché:

```
dotnet restore --no-cache
```

- Obliga a todas las dependencias a ser resueltas incluso si la última restauración fue exitosa:

```
dotnet restore --force
```

- Restaura dependencias usando los orígenes con error como advertencias:

```
dotnet restore --ignore-failed-sources
```

- Restaura dependencias con un nivel específico de verbosidad:

```
dotnet restore --verbosity {{quiet|minimal|normal|detailed|diagnostic}}
```

# dotnet

Herramienta multiplataforma de línea de comandos para .NET Core.

Algunos subcomandos, como **build**, tienen su propia documentación de uso.

Más información: <https://learn.microsoft.com/dotnet/core/tools>.

- Inicializa un proyecto .NET nuevo:

```
dotnet new {{nombre_de_la_plantilla}}
```

- Restaura los paquetes NuGet:

```
dotnet restore
```

- Compila y ejecuta el proyecto .NET en el directorio actual:

```
dotnet run
```

- Ejecuta una aplicación dotnet empaquetada (solo necesita el entorno en tiempo de ejecución, el resto de los comandos requieren el SDK de .NET Core instalado):

```
dotnet {{ruta/a/la/aplicación.dll}}
```

# du

Uso de disco: estima y resume el uso de espacio en disco de archivos y directorios.

Más información: <https://www.gnu.org/software/coreutils/du>.

- Lista los tamaños de un directorio y sus subdirectorios en las unidades dadas (B/KiB/MiB):

```
du -{{b|k|m}} {{ruta/al/directorio}}
```

- Lista los tamaños de un directorio y sus subdirectorios en formato legible para humanos (es decir, seleccionando automáticamente las unidades apropiadas para cada tamaño):

```
du -h {{ruta/al/directorio}}
```

- Muestra el tamaño de un solo directorio en formato legible para humanos:

```
du -sh {{ruta/al/directorio}}
```

- Lista los tamaños legibles para humanos de un directorio y de todos los archivos y directorios dentro del mismo:

```
du -ah {{ruta/al/directorio}}
```

- Lista los tamaños legibles para humanos de un directorio y sus subdirectorios hasta N niveles de profundidad:

```
du -h --max-depth=N {{ruta/al/directorio}}
```

- Lista el tamaño legible para humanos de todos los archivos `.jpg` en subdirectorios del directorio actual y muestra un total al final:

```
du -ch {{*/*.jpg}}
```

# espanso

Expansor de texto multiplataforma escrito en Rust.

Más información: <https://espanso.org>.

- Comprueba el estado de Espanso:

```
espanso status
```

- Edita la configuración de Espanso:

```
espanso edit config
```

- Instala un paquete desde el hub store (<https://hub.espanso.org/>):

```
espanso install {{nombre_paquete}}
```

- Reinicia Espanso (necesario después de instalar un paquete, útil en caso de fallo):

```
espanso restart
```

# fclones

Eficaz buscador y eliminador de archivos duplicados.

Más información: <https://github.com/pkolaczek/fclones>.

- Busca ficheros duplicados en el directorio actual:

```
fclones group .
```

- Busca archivos duplicados en varios directorios y almacena los resultados en la caché:

```
fclones group --cache {{ruta/a/directorio1 ruta/a/directorio2}}
```

- Busca archivos duplicados solo en el directorio especificado, omitiendo los subdirectorios y guarda los resultados en un archivo:

```
fclones group {{ruta/a/directorio}} --depth 1 > {{ruta/a/archivo.txt}}
```

- Mueve los archivos duplicados en un archivo de texto a un directorio diferente:

```
fclones move {{ruta/a/directorio_objetivo}} < {{ruta/a/archivo.txt}}
```

- Simula un enlace simbólico a un archivo de texto sin realmente enlazarlo:

```
fclones link --soft < {{ruta/a/archivo.txt}} --dry-run 2> /dev/null
```

- Elimina los archivos duplicados más recientes en el directorio actual sin almacenarlos en un archivo:

```
fclones group . | fclones remove --priority newest
```

- Preprocesa los archivos JPEG en el directorio actual utilizando un comando externo para eliminar sus datos EXIF antes de buscar duplicados:

```
fclones group . --name '*.jpg' -i --transform 'exiv2 -d a $IN' --in-place
```

# feh

Utilidad ligera de visualización de imágenes.

Más información: <https://feh.finalrewind.org>.

- Muestra imágenes localmente o usando una URL:

```
feh {{ruta/a/imagen}}
```

- Muestra imágenes recursivamente:

```
feh --recursive {{ruta/a/directorio}}
```

- Muestra imágenes sin bordes:

```
feh --borderless {{ruta/a/imagen}}
```

- Cierra después de la última imagen:

```
feh --cycle-once {{ruta/a/imagen}}
```

- Agrega una demora al ciclo de la presentación:

```
feh --slideshow-delay {{segundos}} {{ruta/a/imagen}}
```

- Cambia el fondo de pantalla (centrado, llenar, maximizado, ampliado o amontonado):

```
feh --bg-{{center|fill|max|scale|tile}} {{ruta/a/imagen}}
```

- Crea un montaje de todas las imágenes en un directorio. Produce una nueva imagen:

```
feh --montage --thumb-height {{150}} --thumb-width {{150}} --  
index-info "{{%nn%wx%h}}" --output {{ruta/a/nueva_imagen}}
```

# figlet

Genera encabezados usando caracteres ASCII desde la entrada del usuario.

Vea también **showfigfonts**.

Más información: <http://www.figlet.org/figlet-man.html>.

- Genera el encabezado directamente introduciendo el texto:

```
figlet {{texto_de_entrada}}
```

- Usa un archivo de [f]uente personalizada:

```
figlet {{texto_de_entrada}} -f {{ruta/al/  
archivo_de_fuente.flf}}
```

- Usa una [f]uente del directorio predeterminado (la extensión puede ser omitida):

```
figlet {{texto_de_entrada}} -f {{archivo_de_fuente}}
```

- Redirige la salida de un comando hacia FIGlet:

```
{{comando}} | figlet
```

- Muestra las fuentes de FIGlet disponibles:

```
showfigfonts {{texto_opcional_para_mostrar}}
```

- Utiliza el ancho total del [t]erminal y [c]entra el texto de entrada:

```
figlet -t -c {{texto_de_entrada}}
```

- Muestra todos los caracteres utilizando todo su ancho para evitar traslapes:

```
figlet -W {{input_text}}
```



# fortune

Imprime por pantalla una cita aleatoria (al estilo de una galleta de la suerte).

Más información: <https://manned.org/fortune>.

- Imprime por pantalla una cita:

```
fortune
```

- Imprime por pantalla una cita ofensiva:

```
fortune -o
```

- Imprime por pantalla una cita larga:

```
fortune -l
```

- Imprime por pantalla una cita corta:

```
fortune -s
```

- Muestra una lista de los archivos de citas disponibles:

```
fortune -f
```

- Imprime por pantalla una cita de uno de los archivos mostrados en `fortune -f`:

```
fortune {{archivo}}
```

# fossil ci

Este comando es un alias de **fossil commit**.

Más información: <https://fossil-scm.org/home/help/commit>.

- Muestra la documentación del comando original:

```
tldr fossil-commit
```

# fossil delete

Este comando es un alias de **fossil rm**.

Más información: <https://fossil-scm.org/home/help/delete>.

- Muestra la documentación del comando original:

```
tldr fossil rm
```

# fossil forget

Este comando es un alias de **fossil rm**.

Más información: <https://fossil-scm.org/home/help/forget>.

- Muestra la documentación del comando original:

```
tldr fossil rm
```

# fossil new

Este comando es un alias de **fossil init**.

Más información: <https://fossil-scm.org/home/help/new>.

- Muestra la documentación del comando original:

```
tldr fossil-init
```

# frp

Fast Reverse Proxy: configura rápidamente túneles de red para exponer determinados servicios a Internet o a otras redes externas.

Más información: <https://github.com/fatedier/frp>.

- Vea documentación de `frpc`, el componente cliente `frp`:

`tldr frpc`

- Vea documentación de `frps`, el componente servidor `frp`:

`tldr frps`

# frpc

Conéctate a un servidor **frps** para iniciar conexiones proxy en el host actual.

Parte de **frp**.

Más información: <https://github.com/fatedier/frp>.

- Inicia el servicio, utilizando el archivo de configuración por defecto (se supone que es `frps.ini` en el directorio actual):

```
frpc
```

- Inicia el servicio, utilizando el nuevo archivo de configuración TOML (`frps.toml` en lugar de `frps.ini`) en el directorio actual:

```
frpc {{-c|--config}} ./frps.toml
```

- Inicia el servicio, utilizando un archivo de configuración específico:

```
frpc {{-c|--config}} {{ruta/al/archivo}}
```

- Comprueba si el archivo de configuración es válido:

```
frpc verify {{-c|--config}} {{ruta/al/archivo}}
```

- Imprime script de configuración de autocompletado para Bash, fish, PowerShell o Zsh:

```
frpc completion {{bash|fish|powershell|zsh}}
```

- Muestra versión:

```
frpc {{-v|--version}}
```

# frps

Configura rápidamente un servicio de proxy inverso.

Parte de **frp**.

Más información: <https://github.com/fatedier/frp>.

- Inicia el servicio, utilizando el archivo de configuración por defecto (se supone que es `frps.ini` en el directorio actual):

```
frps
```

- Inicia el servicio, utilizando el nuevo archivo de configuración TOML (`frps.toml` en lugar de `frps.ini`) en el directorio actual:

```
frps {{-c|--config}} ./frps.toml
```

- Inicia el servicio, utilizando un archivo de configuración especificado:

```
frps {{-c|--config}} {{ruta/al/archivo}}
```

- Comprueba si el archivo de configuración es válido:

```
frps verify {{-c|--config}} {{ruta/al/archivo}}
```

- Imprime script de configuración de autocompletado para Bash, fish, PowerShell o Zsh:

```
frps completion {{bash|fish|powershell|zsh}}
```

- Muestra versión:

```
frps {{-v|--version}}
```



# fvm

Gestor de versiones de Flutter.

Más información: <https://fvm.app/documentation/guides/basic-commands>.

- Instala una versión del SDK de Flutter. Invoca el programa sin el argumento `versión` para la configuración del proyecto:

```
fvm install {{versión}}
```

- Establece una versión específica del Flutter SDK en un proyecto:

```
fvm use {{versión}} {{opciones}}
```

- Establece una versión global del SDK de Flutter:

```
fvm global {{versión}}
```

- Borra la caché de FVM:

```
fvm destroy
```

- Elimina una versión específica del SDK de Flutter:

```
fvm remove {{versión}}
```

- Lista todas las versiones instaladas del SDK de Flutter:

```
fvm list
```

- Lista todas las versiones del SDK de Flutter:

```
fvm releases
```

# Gammastep

Ajusta la temperatura del color de la pantalla según la hora del día.

Más información: <https://gitlab.com/chinstrap/gammastep>.

- Activa Gammastep con una [t]emperatura específica durante el día (por ejemplo, 5700k) y por la noche (por ejemplo, 3600k):

```
gammastep -t {{5700}}:{{3600}}
```

- Activa Gammastep con una [l]ocación personalizada especificada manualmente:

```
gammastep -l {{latitud}}:{{longitud}}
```

- Activa Gammastep con un [b]rillo de pantalla específico durante el día (por ejemplo, 70%) y la noche (por ejemplo, 40%), con un brillo mínimo del 10% y uno máximo del 100%:

```
gammastep -b {{0.7}}:{{0.4}}
```

- Activa Gammastep con niveles de [g]ama personalizados (entre 0 y 1):

```
gammastep -g {{rojo}}:{{verde}}:{{azul}}
```

- Activa Gammastep con una temperatura de color c[O]nstante e invariable:

```
gammastep -O {{temperatura}}
```

- Restablece los ajustes de temperatura aplicados por Gammastep:

```
gammastep -x
```

# gau

Obtén todas las URLs: obtén las URLs conocidas de Open Threat Exchange de AlienVault, Wayback Machine y Common Crawl para cualquier dominio.

Más información: <https://github.com/lc/gau>.

- Obtén todas las URLs de un dominio de Open Threat Exchange de AlienVault, Wayback Machine, Common Crawl y URLScan:

```
gau {{ejemplo.com}}
```

- Obtén URLs de varios dominios:

```
gau {{dominio1 dominio2 ...}}
```

- Obtén todas las URLs de varios dominios en un archivo de entrada, ejecutando varios subprocesos:

```
gau --threads {{4}} < {{ruta/a/dominios.txt}}
```

- Escribe los resultados en un archivo:

```
gau {{ejemplo.com}} --o {{ruta/a/urls_encontradas.txt}}
```

- Busca las URLs de un solo proveedor específico:

```
gau --providers {{wayback|commoncrawl|otx|urlscan}}  
{{ejemplo.com}}
```

- Busca las URLs de varios proveedores:

```
gau --providers {{wayback,otx,...}} {{ejemplo.com}}
```

- Busca las URLs dentro de un intervalo de fechas específico:

```
gau --from {{AAAAMM}} --to {{YYYYMM}} {{ejemplo.com}}
```

# gcloud

La herramienta CLI oficial de Google Cloud Platform.

Más información: <https://cloud.google.com/sdk/gcloud>.

- Lista todas las propiedades de la configuración activa:

```
gcloud config list
```

- Inicia sesión en la cuenta de Google:

```
gcloud auth login
```

- Establece como proyecto activo:

```
gcloud config set project {{nombre_del_proyecto}}
```

- SSH en una instancia de máquina virtual:

```
gcloud compute ssh {{usuario}}@{{instancia}}
```

- Muestra todas las instancias de Google Compute Engine de un proyecto. Por defecto, se muestran las instancias de todas las zonas:

```
gcloud compute instances list
```

- Actualiza un archivo kubeconfig con las credenciales adecuadas para apuntar kubectl a un clúster específico en Google Kubernetes Engine:

```
gcloud container clusters get-credentials {{nombre_cluster}}
```

- Actualiza todos los componentes de la CLI de gcloud:

```
gcloud components update
```

- Muestra la ayuda para un comando determinado:

```
gcloud help {{comando}}
```

# gdown

Descarga archivos desde Google Drive y otras URLs.

Más información: <https://github.com/wkentaro/gdown>.

- Descarga un archivo desde una URL:

```
gdown {{url}}
```

- Descarga usando un ID de archivo:

```
gdown {{id_de_archivo}}
```

- Descarga con extracción de ID de archivo difuso (también funciona con enlaces <https://docs.google.com>):

```
gdown --fuzzy {{url}}
```

- Descarga una carpeta utilizando su ID o la URL completa:

```
gdown {{id_de_carpeta|url}} -O {{ruta/a/directorio_de_salida}} --folder
```

- Descarga un archivo tar, escríbelo en stdout y extráelo:

```
gdown {{tar_url}} -O - --quiet | tar xvf -
```

# gh cs

Este comando es un alias de **gh codespace**.

Más información: [https://cli.github.com/manual/gh\\_codespace](https://cli.github.com/manual/gh_codespace).

- Muestra la documentación del comando original:

```
tldr gh-codespace
```

# git add

Añade los archivos cambiados al índice.

Más información: <https://git-scm.com/docs/git-add>.

- Añade un archivo al índice:

```
git add {{ruta/al/archivo}}
```

- Añade todos los archivos (rastreados o no rastreados):

```
git add -A
```

- Añade los archivos ya rastreados:

```
git add -u
```

- Añade también los archivos ignorados:

```
git add -f
```

- Añade partes de archivos interactivamente:

```
git add -p
```

- Añade partes de un archivo dado interactivamente:

```
git add -p {{ruta/al/archivo}}
```

- Añade un archivo interactivamente:

```
git add -i
```

# git am

Aplica archivos de parche. Útil cuando se reciben commits por correo electrónico.

Vea también **git format-patch**, comando que genera archivos de parche.

Más información: <https://git-scm.com/docs/git-am>.

- Aplica un archivo de parche:

```
git am {{ruta/al/archivo.patch}}
```

- Aplica un archivo de parche remoto:

```
curl -L {{https://ejemplo.com/file.patch}} | git apply
```

- Aborta el proceso de aplicar un archivo de parche:

```
git am --abort
```

- Aplica todo lo posible de un archivo de parche y guarda los fragmentos fallidos para rechazar archivos:

```
git am --reject {{ruta/al/archivo.patch}}
```



# git bisect

Utiliza la búsqueda binaria para encontrar la confirmación que introdujo un error.

Git salta de un lado a otro del gráfico de confirmaciones hasta alcanzar progresivamente la confirmación defectuosa.

Más información: <https://git-scm.com/docs/git-bisect>.

- Comienza una sesión de bisecado en un rango de confirmaciones delimitado por una confirmación errónea conocida y por una sana conocida (normalmente más antigua):

```
git bisect start {{confirmación_errónea}}  
{{confirmación_buena}}
```

- Para cada confirmación que `git bisect` seleccione, marca como "mala" (bad) o "buena" (good) después de probarla para el problema:

```
git bisect {{good|bad}}
```

- Termina la sesión de bisecado y vuelve a la rama anterior después de que `git-bisect` determine con precisión la confirmación defectuosa:

```
git bisect reset
```

- Omite una confirmación durante una sesión de bisecado (p. ej. una que falla las pruebas debido a un problema diferente):

```
git bisect skip
```

- Muestra un registro de lo que se ha hecho hasta el momento:

```
git bisect log
```

# git blame

Muestra el hash de la confirmación y el último autor de cada línea de un archivo.

Más información: <https://git-scm.com/docs/git-blame>.

- Muestra el archivo con el nombre del autor y el hash de la confirmación en cada línea:

```
git blame {{archivo}}
```

- Muestra el archivo con correo electrónico del autor y hash de la confirmación en cada línea:

```
git blame {{-e|--show-email}} {{archivo}}
```

- Muestra quien y que modificó de un archivo, a partir de una confirmación específica:

```
git blame {{confirmación}} {{ruta/al/archivo}}
```

- Muestra quien y que modificó de un archivo, a partir de la confirmación anterior a otra:

```
git blame {{confirmación}}~ {{ruta/al/archivo}}
```

# git branch

Comando principal de Git para trabajar con ramas.

Más información: <https://git-scm.com/docs/git-branch>.

- Lista todas las ramas (locales y remotas; la rama actual se resalta con \*):

```
git branch --all
```

- Lista las ramas que incluyen una confirmación específica en su historial:

```
git branch --all --contains {{hash_de_la_confirmación}}
```

- Muestra el nombre de la rama actual:

```
git branch --show-current
```

- Crea una nueva rama basada en la confirmación actual:

```
git branch {{nombre_rama}}
```

- Crea una nueva rama basada en una confirmación específica:

```
git branch {{nombre_de_rama}} {{hash_de_la_confirmación}}
```

- Renombra una rama (para ello no debes tenerla controlada):

```
git branch {{-m|--move}} {{nombre_de_rama_antigua}}  
{{nuevo_nombre_rama}}
```

- Elimina una rama local (no debes tenerla controlada para hacerlo):

```
git branch {{-d|--delete}} {{nombre_de_rama}}
```

- Elimina una rama remota:

```
git push {{nombre_remoto}} --delete {{nombre_de_rama_remota}}
```

# git bundle

Empaqueta objetos y referencias en un archivo.

Más información: <https://git-scm.com/docs/git-bundle>.

- Crea un archivo bundle que contiene todos los objetos y referencias de una rama específica:

```
git bundle create {{ruta/al/archivo.bundle}} {{nombre_rama}}
```

- Crea un archivo bundle de todas las ramas:

```
git bundle create {{ruta/al/archivo.bundle}} --all
```

- Crea un archivo bundle de los últimos 5 commits de la rama actual:

```
git bundle create {{ruta/al/archivo.bundle}} -5 {{HEAD}}
```

- Crea un archivo bundle de los últimos 7 días:

```
git bundle create {{ruta/al/archivo.bundle}} --since 7.days  
{{HEAD}}
```

- Verifica que un archivo bundle es válido y puede aplicarse al repositorio actual:

```
git bundle verify {{ruta/al/archivo.bundle}}
```

- Imprime en stdout la lista de referencias contenidas en un bundle:

```
git bundle unbundle {{ruta/al/archivo.bundle}}
```

- Desagrupa una rama específica de un archivo bundle en el repositorio actual:

```
git pull {{ruta/al/archivo.bundle}} {{nombre_rama}}
```

- Crea un nuevo repositorio a partir de un paquete:

```
git clone {{ruta/al/archivo.bundle}}
```

# git check-ignore

Analiza y depura los archivos que Git debe ignorar / excluir (.gitignore).

Más información: <https://git-scm.com/docs/git-check-ignore>.

- Comprueba si un archivo o directorio es ignorado:

```
git check-ignore {{ruta/al/archivo_o_directorio}}
```

- Comprueba si varios archivos o directorios son ignorados:

```
git check-ignore {{ruta/al/archivo_o_directorio1 ruta/al/archivo_o_directorio2 ...}}
```

- Usa nombres de rutas, uno por línea, a partir de la entrada estandar (stdin):

```
git check-ignore --stdin < {{ruta/al/archivo_lista}}
```

- Comprueba sin leer el índice (se utiliza para depurar por qué las rutas fueron rastreadas y no ignoradas):

```
git check-ignore --no-index {{ruta/al/archivo_o_directorio1 ruta/al/archivo_o_directorio2 ...}}
```

- Incluye detalles sobre el patrón de coincidencia para cada ruta:

```
git check-ignore --verbose {{ruta/al/archivo_o_directorio1 ruta/al/archivo_o_directorio2 ...}}
```

# git checkout

Comprueba una rama o rutas con el árbol de trabajo.

Más información: <https://git-scm.com/docs/git-checkout>.

- Crea una nueva rama y se cambia a la misma:

```
git checkout -b {{nombre_de_la_rama}}
```

- Crea una nueva rama a partir de una referencia específica (rama, remoto/ rama, las etiquetas son ejemplos de referencias válidas) y cambiarse a esta:

```
git checkout -b {{nombre_de_la_rama}} {{referencia}}
```

- Cambia a una rama local existente:

```
git checkout {{nombre_de_la_rama}}
```

- Cambia a la rama previamente comprobada:

```
git checkout -
```

- Cambia a una rama remota existente:

```
git checkout --track {{nombre_remoto}}/{{nombre_de_la_rama}}
```

- Descarta todos los cambios sin marcar en el directorio actual (vea `git reset` para más comandos para deshacer):

```
git checkout .
```

- Descarta los cambios no marcados de un archivo específico:

```
git checkout {{nombre_del_archivo}}
```

- Reemplaza un archivo en el directorio actual con la versión de este en la confirmación de una rama específica:

```
git checkout {{nombre_de_la_rama}} -- {{nombre_del_archivo}}
```

# git cherry-pick

Aplica los cambios introducidos por confirmaciones existentes a la rama actual.

Para aplicar cambios a otra rama, primero utiliza **git checkout** para cambiar a la rama deseada.

Más información: <https://git-scm.com/docs/git-cherry-pick>.

- Aplica una confirmación a la rama actual:

```
git cherry-pick {{confirmación}}
```

- Aplica un rango de confirmaciones de la rama actual (vea también `git rebase --onto`):

```
git cherry-pick {{confirmación_inicial}}~..  
{{confirmación_final}}
```

- Aplica múltiples confirmaciones no secuenciales a la rama actual:

```
git cherry-pick {{confirmación_1 confirmación_2 ...}}
```

- Añade los cambios de una confirmación al directorio de trabajo, sin crear una confirmación:

```
git cherry-pick --no-commit {{confirmación}}
```

# git clean

Elimina archivos sin rastrear del árbol de trabajo.

Más información: <https://git-scm.com/docs/git-clean>.

- Elimina archivos que no son rastreados por Git:

```
git clean
```

- Elimina interactivamente archivos que no son rastreados por Git:

```
git clean {{-i|--interactive}}
```

- Muestra que archivos serían borrados sin llegar a borrarlos:

```
git clean --dry-run
```

- Elimina forzosamente los archivos que no son rastreados por Git:

```
git clean {{-f|--force}}
```

- Elimina forzosamente los directorios que no son rastreados por Git:

```
git clean {{-f|--force}} -d
```

- Elimina archivos sin rastrear, incluyendo los archivos ignorados en `.gitignore` y los excluidos en `.git/info/exclude`:

```
git clean -x
```



# git clone

Clona un repositorio existente.

Más información: <https://git-scm.com/docs/git-clone>.

- Clona un repositorio existente en un directorio nuevo (el directorio por defecto es el nombre del repositorio):

```
git clone {{ubicación_repositorio_remoto}} {{ruta/al/directorio}}
```

- Clona un repositorio existente y sus submódulos:

```
git clone --recursive {{ubicación_repositorio_remoto}}
```

- Clona sólo el directorio `.git` de un repositorio existente:

```
git clone --no-checkout {{ubicación_repositorio_remoto}}
```

- Clona un repositorio local:

```
git clone --local {{ruta/al/repositorio/local}}
```

- Clona en silencio:

```
git clone --quiet {{ubicación_repositorio_remoto}}
```

- Clona un repositorio existente obteniendo sólo los 10 commits más recientes de la rama por defecto (útil para ahorrar tiempo):

```
git clone --depth {{10}} {{ubicación_repositorio_remoto}}
```

- Clona un repositorio existente buscando sólo una rama específica:

```
git clone --branch {{nombre}} --single-branch  
{{ubicación_repositorio_remoto}}
```

- Clona un repositorio existente usando un comando SSH específico:

```
git clone --config core.sshCommand="{{ssh -i ruta/a/  
clave_privada_ssh}}" {{ubicación_repositorio_remoto}}
```

# git commit

Realiza confirmaciones de los archivos al repositorio.

Más información: <https://git-scm.com/docs/git-commit>.

- Realiza una confirmación de los archivos marcados al repositorio con un mensaje:

```
git commit --message "{{mensaje}}"
```

- Realiza una confirmación de los archivos marcados con un mensaje leído desde un archivo:

```
git commit --file {{ruta/al/  
archivo_con_mensaje_de_la_confirmación}}
```

- Marca automáticamente todos los archivos modificados y realiza una confirmación con un mensaje:

```
git commit --all --message "{{mensaje}}"
```

- Confirma todos los archivos preparados y los firma con una llave de GPG (o la llave en el archivo de configuración si no se especifica un argumento):

```
git commit --gpg-sign {{identificador_de_llave}} --message  
"{{mensaje}}"
```

- Sustituye la última confirmación con los cambios marcados actualmente, cambiando el hash de la confirmación:

```
git commit --amend
```

- Realiza una confirmación para archivos específicos (marcados previamente):

```
git commit {{ruta/al/archivo1 ruta/al/archivo2 ...}}
```

- Crea una confirmación, incluso si no hay archivos marcados:

```
git commit --message "{{mensaje}}" --allow-empty
```

# git config

Gestiona opciones de configuración personalizadas para repositorios Git.

Estas configuraciones pueden ser locales (para el repositorio actual) o globales (para el usuario actual).

Más información: <https://git-scm.com/docs/git-config>.

- Establece globalmente tu nombre o correo electrónico (esta información es necesaria para hacer un commit en un repositorio y se incluirá en todos los commits):

```
git config --global {{user.name|user.email}} "{{{Tu nombre|email@example.com}}}"
```

- Lista las entradas de configuración local o global:

```
git config --list --{{local|global}}
```

- Lista sólo las entradas de configuración del sistema (almacenadas en `/etc/gitconfig`), y muestra la ubicación de dicho archivo:

```
git config --list --system --show-origin
```

- Obtén el valor de una entrada de configuración dada:

```
git config alias.unstage
```

- Establece el valor global de una entrada de configuración dada:

```
git config --global alias.unstage "reset HEAD --"
```

- Revierte una entrada de configuración global a su valor por defecto:

```
git config --global --unset alias.unstage
```

- Edita la configuración local de Git (`.git/config`) en el editor por defecto:

```
git config --edit
```

- Edita la configuración global de Git (`~/.gitconfig` por defecto o `$XDG_CONFIG_HOME/git/config` si existe tal archivo) en el editor por defecto:

```
git config --global --edit
```

# git diff

Muestra los cambios de los archivos rastreados.

Más información: <https://git-scm.com/docs/git-diff>.

- Muestra los cambios sin marcar ni confirmación:

```
git diff
```

- Muestra todos los cambios sin confirmación, pero incluye los marcados:

```
git diff HEAD
```

- Muestra solo los cambios marcados pero que no tienen confirmación:

```
git diff --staged
```

- Muestra los cambios de todas las confirmaciones a partir de una fecha y/o tiempo específico (p. ej., 1 week 2 days o una fecha ISO):

```
git diff 'HEAD@{3 months|weeks|days|hours|seconds ago}'
```

- Muestra solo los nombres de los archivos cambiados en una confirmación específica:

```
git diff --name-only {{confirmación}}
```

- Muestra un resumen de los cambios hecho en una confirmación (p. ej. permisos de un archivo):

```
git diff --summary {{confirmación}}
```

- Compara un único archivo entre dos ramas o confirmaciones:

```
git diff {{rama_1}}..{{rama_2}} [--] {{ruta/al/archivo}}
```

- Compara diferentes archivos de la rama actual con otra rama:

```
git diff {{rama}}:{{ruta/al/archivo}} {{ruta/al/archivo2}}
```

# git fetch

Descarga objetos y referencias de un repositorio remoto.

Más información: <https://git-scm.com/docs/git-fetch>.

- Recibe los últimos cambios del repositorio remoto upstream por defecto (si se ha establecido):

```
git fetch
```

- Recibe las ramas nuevas de un repositorio remoto upstream específico:

```
git fetch {{remote_name}}
```

- Recibe los últimos cambios de todos los repositorios remotos upstream:

```
git fetch --all
```

- Recibe también las etiquetas de un repositorio upstream:

```
git fetch --tags
```

- Elimina las referencias locales a ramas remotas que han sido eliminadas de upstream:

```
git fetch --prune
```

# git format-patch

Prepara archivos .patch. Es útil cuando se envían commits por correo electrónico.

Vea también **git-am**, comando que puede aplicar los archivos .patch generados.

Más información: <https://git-scm.com/docs/git-format-patch>.

- Crea un archivo .patch con nombre automático para todos los cambios que no están en el push:

```
git format-patch {{origen}}
```

- Escribe un archivo .patch para todos los commits entre dos revisiones a stdout:

```
git format-patch {{revisión_1}}..{{revisión_2}}
```

- Escribe un archivo .patch para los 3 últimos commits:

```
git format-patch -{{3}}
```

# git gc

Optimiza el repositorio local eliminando archivos innecesarios.

Más información: <https://git-scm.com/docs/git-gc>.

- Optimiza el repositorio:

```
git gc
```

- Optimiza agresivamente (tarda más):

```
git gc --aggressive
```

- Optimiza sin eliminar objetos sueltos (por defecto los elimina):

```
git gc --no-prune
```

- Suprime toda la salida:

```
git gc --quiet
```

- Muestra todas sus funcionalidades:

```
git gc --help
```

# git-grep

Encuentra dentro de archivos en cualquier parte del historial del repositorio.

Acepta una gran cantidad de opciones, de la misma manera que el comando **grep**.

Más información: <https://git-scm.com/docs/git-grep>.

- Busca una cadena en los archivos rastreados:

```
git grep {{cadena_a_buscar}}
```

- Busca una cadena en archivos que coincidan con un patrón entre los archivos rastreados:

```
git grep {{cadena_a_buscar}} -- {{patrón_de_archivos}}
```

- Busca una cadena en los archivos rastreados, incluyendo submódulos:

```
git grep --recurse-submodules {{cadena_a_buscar}}
```

- Busca una cadena en un punto específico del historial:

```
git grep {{cadena_a_buscar}} {{HEAD~2}}
```

- Busca una cadena a través de todas las ramas:

```
git grep {{cadena_a_buscar}} $(git rev-list --all)
```



# git gui

Una GUI para Git para gestionar ramas, remotos, confirmaciones de cambio y realizar fusiones locales.

Vea también: **git-cola**, **gitk**.

Más información: <https://git-scm.com/docs/git-gui>.

- Inicia la GUI:

```
git gui
```

- Muestra un archivo específico con el nombre del autor y el hash de confirmación en cada línea:

```
git gui blame {{ruta/al/archivo}}
```

- Abre `git gui blame` en una revisión específica:

```
git gui blame {{revisión}} {{ruta/al/archivo}}
```

- Abre `git gui blame` y desplaza la vista para centrarla en una línea específica:

```
git gui blame --line={{línea}} {{ruta/al/archivo}}
```

- Abre una ventana para hacer una confirmación y vuelve al intérprete de comandos cuando se haya completado:

```
git gui citool
```

- Abre `git gui citool` en modo "Modificar la última confirmación":

```
git gui citool --amend
```

- Abre `git gui citool` en modo de solo lectura:

```
git gui citool --nocommit
```

- Muestra un navegador para el árbol de una rama específica, abriendo la herramienta de autoría al hacer clic en los archivos:

```
git gui browser maint
```

# git-imerge

Ejecuta una fusión o rebase entre dos ramas Git incrementalmente.

Los conflictos entre las ramas se rastrean a pares de commits individuales para simplificar la resolución de conflictos.

Más información: <https://github.com/mhagger/git-imerge>.

- Inicia un rebase de tipo imerge (primero comprueba la rama a ser rebasada):

```
git imerge rebase {{rama_a_rebasar}}
```

- Inicia una fusión de tipo imerge (primero comprueba la rama en la que fusionar):

```
git imerge merge {{rama_a_fusionar}}
```

- Muestra una diagrama ASCII para la fusión o rebase en proceso:

```
git imerge diagram
```

- Continúa la operación imerge después de resolver los conflictos (primero añade con `git add` los archivos conflictivos):

```
git imerge continue --no-edit
```

- Concluye una operación imerge después de que todos los conflictos se hayan resuelto:

```
git imerge finish
```

- Aborta una operación imerge y vuelve a la rama anterior:

```
git-imerge remove && git checkout {{rama_anterior}}
```

# git init

Inicializa un nuevo repositorio Git local.

Más información: <https://git-scm.com/docs/git-init>.

- Inicializa un nuevo repositorio local:

```
git init
```

- Inicializa un repositorio con un nombre específico para la rama inicial:

```
git init --initial-branch={{nombre_de_la_rama}}
```

- Inicializa un repositorio usando SHA256 como hash del objeto (requiere la versión 2.29+ de Git):

```
git init --object-format={{sha256}}
```

- Inicializa un repositorio vacío, adecuado para usarlo como remoto a través de SSH:

```
git init --bare
```

# git lfs

Trabaja con archivos grandes en repositorios de Git.

Más información: <https://git-lfs.com>.

- Inicializa Git LFS:

```
git lfs install
```

- Rastrea archivos que coinciden con un patrón:

```
git lfs track '{{*.bin}}'
```

- Cambia la URL a la que apunta Git LFS (útil si el servidor LFS está separado del servidor Git):

```
git config {{-f|--file}} .lfsconfig lfs.url  
{{url_del_punto_de_acceso_LFS}}
```

- Muestra los patrones rastreados:

```
git lfs track
```

- Muestra los archivos que han sido añadidos con un commit:

```
git lfs ls-files
```

- Introduce todos los objetos LFS en el servidor remoto (útil si se encuentran errores):

```
git lfs push --all {{nombre_remoto}} {{nombre_de_la_rama}}
```

- Trae todos los objetos de Git LFS:

```
git lfs fetch
```

- Verifica todos los objetos de Git LFS:

```
git lfs checkout
```

# git log

Muestra un historial de confirmaciones.

Más información: <https://git-scm.com/docs/git-log>.

- Muestra la secuencia de confirmaciones comenzando desde el actual, en orden cronológico inverso, del repositorio de Git en el directorio de trabajo actual:

```
git log
```

- Muestra el historial de un archivo o directorio específico, incluyendo las diferencias:

```
git log {{-p|-u|--patch}} {{ruta/al/archivo_o_directorio}}
```

- Muestra un resumen de los archivos, o archivo, cambiados en cada confirmación:

```
git log --stat
```

- Muestra un gráfico de las confirmaciones en la rama actual, utilizando solo la primera línea del mensaje de cada uno:

```
git log --oneline --graph
```

- Muestra un gráfico de todas las confirmaciones, etiquetas y ramas en todo el repositorio:

```
git log --oneline --decorate --all --graph
```

- Muestra solo las confirmaciones cuyo mensaje incluye una cadena dada (no diferencia entre mayúsculas y minúsculas):

```
git log {{-i|--regexp-ignore-case}} --grep  
{{cadena_a_buscar}}
```

- Muestra las últimas N confirmaciones de determinado autor:

```
git log {{-n|--max-count}} {{número}} --author "{{autor}}"
```

- Muestra las confirmaciones entre dos fechas (yyyy-mm-dd):

```
git log --before "{{2017-01-29}}" --after "{{2017-01-17}}"
```

# git ls-tree

Muestra los contenidos de un objeto árbol.

Más información: <https://git-scm.com/docs/git-ls-tree>.

- Lista el contenido del árbol en una rama:

```
git ls-tree {{nombre_de_la_rama}}
```

- Lista el contenido del árbol en una confirmación (recursivo en subárboles):

```
git ls-tree -r {{hash_de_la_confirmación}}
```

- Lista solo los nombres de archivos del árbol en una confirmación:

```
git ls-tree --name-only {{hash_de_la_confirmación}}
```

- Lista el nombre de los archivos en la rama actual en forma de árbol (Nota: la opción `--fromfile` no está disponible en el comando `tree` de Windows):

```
git ls-tree -r --name-only HEAD | tree --fromfile
```

# git merge

Fusiona ramas.

Más información: <https://git-scm.com/docs/git-merge>.

- Fusiona una rama con la rama actual:

```
git merge {{nombre_de_la_rama}}
```

- Edita el mensaje de fusión:

```
git merge --edit {{nombre_de_la_rama}}
```

- Fusiona una rama y crea una confirmación para la fusión:

```
git merge --no-ff {{nombre_de_la_rama}}
```

- Cancela una fusión en caso de conflictos:

```
git merge --abort
```

- Fusiona usando una estrategia específica:

```
git merge --strategy {{estrategia}} --strategy-option  
{{opción_de_estrategia}} {{nombre_de_la_rama}}
```

# git mktree

Construye un objeto árbol usando texto formateado **ls-tree**.

Más información: <https://git-scm.com/docs/git-mktree>.

- Construye un objeto árbol y verifica que el hash de cada entrada del árbol identifica un objeto existente:

```
git mktree
```

- Permite que falten objetos:

```
git mktree --missing
```

- Lee la salida terminada en NUL (carácter cero) del objeto árbol (**ls-tree -z**):

```
git mktree -z
```

- Permite la creación de múltiples objetos árbol:

```
git mktree --batch
```

- Ordena y construye un árbol a partir de **stdin** (se requiere un formato de salida de **git ls-tree** no recursivo):

```
git mktree < {{ruta/a/árbol.txt}}
```



# git mv

Mueve o renombra archivos y actualiza el índice Git.

Más información: <https://git-scm.com/docs/git-mv>.

- Mueve el archivo dentro del repositorio y añade el movimiento al siguiente commit:

```
git mv {{ruta/al/archivo}} {{nueva/ruta/al/archivo}}
```

- Renombra un archivo y añade el renombre al siguiente commit:

```
git mv {{nombre_de_archivo}} {{nuevo_nombre_de_archivo}}
```

- Sobrescribe el archivo en la ruta objetivo si existe:

```
git mv --force {{archivo}} {{objetivo}}
```

# git pr

Comprueba las solicitudes de extracción de cambios (pull requests) de GitHub localmente.

Más información: <https://github.com/tj/git-extras/blob/master/Commands.md#git-pr>.

- Comprueba un pull request específica:

```
git pr {{número_pr}}
```

- Comprueba un pull request para un remoto específico:

```
git pr {{número_pr}} {{remoto}}
```

- Comprueba un pull request a partir de su URL:

```
git pr {{url}}
```

- Limpia las ramas antiguas de pull requests:

```
git pr clean
```

# git pull

Obtiene rama de un repositorio remoto y lo fusiona con el repositorio local.

Más información: <https://git-scm.com/docs/git-pull>.

- Descarga cambios del repositorio remoto por defecto y lo fusiona:

```
git pull
```

- Descarga cambios del repositorio remoto por defecto y usa avance rápido (fast forward):

```
git pull --rebase
```

- Descarga cambios de un repositorio remoto y una rama específica para fusionarlos en HEAD:

```
git pull {{nombre_remoto}} {{rama}}
```

# git push

Envía (push) los commits al repositorio remoto.

Más información: <https://git-scm.com/docs/git-push>.

- Envía los cambios locales en la rama actual a la misma rama en el remoto:

```
git push
```

- Envía los cambios locales de una rama específica a la misma rama en el remoto:

```
git push {{nombre_remoto}} {{rama_local}}
```

- Publica la rama actual en el repositorio remoto y establece el nombre remoto de la rama:

```
git push -u {{nombre_remoto}} {{rama_remota}}
```

- Envía los cambios locales de una rama específica a una rama específica en el remoto:

```
git push {{nombre_remoto}} {{rama_local}}:{{rama_remota}}
```

- Envía los cambios de todas las ramas locales a sus respectivas ramas en el repositorio remoto:

```
git push --all {{nombre_remoto}}
```

- Elimina una rama en el repositorio remoto:

```
git push {{nombre_remoto}} --delete {{rama_remota}}
```

- Elimina las ramas remotas que no están en el repositorio local:

```
git push --prune {{nombre_remoto}}
```

- Publica las etiquetas que aún no están en el repositorio remoto:

```
git push --tags
```

# git rebase

Vuelve a aplicar confirmaciones de una rama en lo más alto de otra rama.

Se utiliza comúnmente para "mover" una rama entera a otra base, ya que crea copias de las confirmaciones en una nueva ubicación.

Más información: <https://git-scm.com/docs/git-rebase>.

- Reorganiza la rama actual en lo más alto de otra rama:

```
git rebase {{nueva_base_rama}}
```

- Inicia un rebase interactivo que permite reordenar, omitir, combinar o modificar confirmaciones:

```
git rebase {{-i|--interactive}}  
{{rama_base_objetivo_o_hash_de_la_confirmación}}
```

- Continúa un rebase que fue interrumpido por una fusión fallida después de editar los archivos con conflictos:

```
git rebase --continue
```

- Continúa un rebase que fue pausado para fusionar conflictos saltando la confirmación conflictiva:

```
git rebase --skip
```

- Cancela un rebase en proceso (por ej., si es interrumpido por un conflicto de fusión):

```
git rebase --abort
```

- Mueve parte de la rama actual a una nueva base proporcionando la base antigua para empezar:

```
git rebase --onto {{base_nueva}} {{base_antigua}}
```

- Reaplica las últimas cinco confirmaciones en su lugar, evita que puedan ser reordenadas, omitidas, combinadas o modificadas:

```
git rebase {{-i|--interactive}} {{HEAD~5}}
```

- Resuelve automáticamente cualquier conflicto favoreciendo la versión de la rama en la que se está trabajando (en este caso la palabra `theirs` tiene un significado invertido):

```
git rebase {{-X|--strategy-option}} theirs {{nombre_rama}}
```

# git reflog

Muestra un registro de cambios de las referencias (reflog) locales como HEAD, ramas o etiquetas.

Más información: <https://git-scm.com/docs/git-reflog>.

- Muestra un registro de referencias para HEAD:

```
git reflog
```

- Muestra el registro de referencias para una rama:

```
git reflog {{nombre_de_la_rama}}
```

- Muestra solo las últimas 5 entradas en el registro de referencias:

```
git reflog {{-n|-max-count}} 5
```

# git remote

Gestiona el conjunto de repositorios rastreados ("remotos").

Más información: <https://git-scm.com/docs/git-remote>.

- Muestra una lista de los remotos existentes, sus nombres y URL:

```
git remote {{-v|--verbose}}
```

- Muestra información de un remoto:

```
git remote show {{nombre_remoto}}
```

- Añade un remoto:

```
git remote add {{nombre_remoto}} {{url_remoto}}
```

- Cambia la URL de un remoto (utiliza - -add para mantener la URL existente):

```
git remote set-url {{nombre_remoto}} {{nueva_url}}
```

- Muestra la URL de un remoto:

```
git remote get-url {{nombre_del_remoto}}
```

- Elimina un remoto:

```
git remote remove {{nombre_remoto}}
```

- Renombra un remoto:

```
git remote rename {{nombre_antiguo}} {{nombre_nuevo}}
```



# git reset

Deshaz confirmaciones o desmarca cambios mediante el restablecimiento del actual HEAD de Git al estado especificado.

Si se pasa una ruta, funciona como "desmarcar", si se pasa el hash de una confirmación o una rama, funciona como "deshacer" la confirmación.

Más información: <https://git-scm.com/docs/git-reset>.

- Desmarca todo:

```
git reset
```

- Desmarca un archivo o archivos específicos:

```
git reset {{ruta/al/archivo_o_archivos}}
```

- Interactivamente desmarca partes de un archivo:

```
git reset --patch {{ruta/al/archivo}}
```

- Deshaz la última confirmación, manteniendo sus cambios, y cualquier otro cambio sin confirmación, en el sistema de archivos:

```
git reset HEAD~
```

- Deshaz las últimas dos confirmaciones al añadir sus cambios al índice (p. ej. marcado para confirmación):

```
git reset --soft HEAD~2
```

- Descarta cualquier cambio sin confirmación, marcado o no (se puede `git checkout` solo para los cambios sin marcar):

```
git reset --hard
```

- Restablece el repositorio a una confirmación específica y descarta a partir de esta los cambios con y sin confirmación, y los marcados:

```
git reset --hard {{confirmación}}
```

# git restore

Restaura los archivos del árbol de trabajo. Requiere la version 2.23+ de Git.

Vea también **git checkout** y **git reset**.

Más información: <https://git-scm.com/docs/git-restore>.

- Restaura un archivo sin marcar a la versión de la confirmación actual (HEAD):

```
git restore {{ruta/al/archivo}}
```

- Restaura un archivo sin marcar a la versión de una confirmación específica:

```
git restore --source {{confirmación}} {{ruta/al/archivo}}
```

- Descarta los cambios sin confirmación para los archivos rastreados:

```
git restore :/
```

- Desmarca un archivo:

```
git restore --staged {{ruta/al/archivo}}
```

- Desmarca todos los archivos:

```
git restore --staged :/
```

- Descarta todos los cambios de los archivos, marcados o no:

```
git restore --worktree --staged :/
```

- Selecciona interactivamente secciones de archivos para restaurar:

```
git restore --patch
```

# git rev-list

Muestra las revisiones (confirmaciones) en orden cronológico inverso.

Más información: <https://git-scm.com/docs/git-rev-list>.

- Muestra todas las confirmaciones de la rama actual:

```
git rev-list {{HEAD}}
```

- Imprime la última confirmación que cambió (agregó/editó/eliminó) un archivo específico en la rama actual:

```
git rev-list {{-n|--max-count}} 1 HEAD -- {{ruta/al/archivo}}
```

- Muestra las confirmaciones más recientes a partir de una fecha y una rama específica:

```
git rev-list --since "{{2019-12-01 00:00:00}}"  
{{nombre_de_rama}}
```

- Muestra todas las confirmaciones fusionadas en una confirmación específica:

```
git rev-list --merges {{confirmación}}
```

- Imprime el número de confirmaciones desde una etiqueta específica:

```
git rev-list {{nombre_de_la_etiqueta}}..HEAD --count
```

# git rev-parse

Muestra metadatos relativos a revisiones específicas.

Más información: <https://git-scm.com/docs/git-rev-parse>.

- Obtén el hash de la confirmación de una rama:

```
git rev-parse {{nombre_de_la_rama}}
```

- Obtén el nombre de la rama actual:

```
git rev-parse --abbrev-ref {{HEAD}}
```

- Obtén la ruta absoluta al directorio raíz:

```
git rev-parse --show-toplevel
```

# git revert

Crea nuevas confirmaciones que revierten el efecto de los anteriores.

Más información: <https://git-scm.com/docs/git-revert>.

- Revierte la confirmación más reciente:

```
git revert {{HEAD}}
```

- Revierte la quinta confirmación más reciente:

```
git revert HEAD~{{4}}
```

- Revierte una confirmación específica:

```
git revert {{0c01a9}}
```

- Revierte múltiples confirmaciones:

```
git revert {{nombre_rama~5..nombre_rama~2}}
```

- Revierte confirmaciones sin crear nuevas confirmaciones:

```
git revert {{-n|--no-commit}} {{0c01a9..9a1743}}
```

# git rm

Elimina archivos del índice del repositorio y del sistema de archivos local.

Más información: <https://git-scm.com/docs/git-rm>.

- Elimina un archivo del índice de un repositorio y del sistema de archivos local:

```
git rm {{archivo}}
```

- Elimina un directorio:

```
git rm -r {{directorio}}
```

- Elimina un archivo del índice del repositorio, pero mantiene intacto el archivo local:

```
git rm --cached {{archivo}}
```

# git shortlog

Resume la salida de **git log**.

Más información: <https://git-scm.com/docs/git-shortlog>.

- Muestra un resumen de todas las confirmaciones realizadas, agrupadas alfabéticamente por autor:

```
git shortlog
```

- Muestra un resumen de todas las confirmaciones realizadas, agrupadas por el número de confirmaciones realizadas:

```
git shortlog {{-n|--numbered}}
```

- Muestra un resumen de todas las confirmaciones realizadas, agrupadas por la identidad de quien realiza la confirmación (usuario y correo electrónico):

```
git shortlog {{-c|--committer}}
```

- Muestra un resumen de las últimas cinco confirmaciones (i. e., un rango de revisiones específico):

```
git shortlog HEAD~5..HEAD
```

- Muestra todos los usuarios, correos electrónicos y número de confirmaciones en la rama actual:

```
git shortlog {{-s|--summary}} {{-n|--numbered}} {{-e|--email}}
```

- Muestra todos los usuarios, correos electrónicos y número de confirmaciones en todas las ramas:

```
git shortlog {{-s|--summary}} {{-n|--numbered}} {{-e|--email}} --all
```

# git show

Muestra varios tipos de objetos Git (confirmaciones, etiquetas, etcétera).

Más información: <https://git-scm.com/docs/git-show>.

- Muestra información sobre la última confirmación (hash, mensaje, cambios y otros metadatos):

```
git show
```

- Muestra información de una confirmación específica:

```
git show {{confirmación}}
```

- Muestra información de la confirmación asociada a una determinada etiqueta:

```
git show {{etiqueta}}
```

- Muestra información de la tercera confirmación desde la punta de una rama:

```
git show {{rama}}~{{3}}
```

- Muestra el mensaje de una confirmación en una única línea, eliminando el resultado de la diferencia:

```
git show --oneline -s {{confirmación}}
```

- Muestra solo estadísticas (caracteres agregados o removidos) de los archivos modificados:

```
git show --stat {{confirmación}}
```

- Muestra solo la lista de archivos agregados, renombrados o eliminados:

```
git show --summary {{confirmación}}
```

- Muestra el contenido de un archivo en una revisión específica (por ej., una rama, una etiqueta o una confirmación):

```
git show {{revisión}}:{{ruta/al/archivo}}
```



# git sizer

Calcula diferentes métricas del repositorio Git y te informa de cualquier que puede causar problemas o inconvenientes.

Más información: <https://github.com/github/git-sizer>.

- Reporta solo estadísticas que tienen un nivel de preocupación mayor que 0:

```
git sizer
```

- Reporta todas las estadísticas:

```
git sizer -v
```

- Muestra opciones adicionales:

```
git sizer -h
```

# git stash

Almacena los cambios locales de Git en un área temporal.

Más información: <https://git-scm.com/docs/git-stash>.

- Almacena los cambios actuales, excepto los archivos nuevos (sin seguimiento):

```
git stash push --message {{mensaje_opcional_stash}}
```

- Almacena los cambios actuales, incluyendo los archivos nuevos (sin seguimiento):

```
git stash --include-untracked
```

- Selecciona interactivamente partes de los archivos modificados para almacenarlos:

```
git stash --patch
```

- Lista todos los stashes (muestra el nombre del stash, la rama relacionada y el mensaje):

```
git stash list
```

- Muestra los cambios como un parche entre el stash (por defecto es `stash@{0}`) y la confirmación de cuando se creó la entrada stash por primera vez:

```
git stash show --patch {{stash@{0}}}
```

- Aplica un stash (por defecto es el último, llamado `stash@{0}`):

```
git stash apply {{nombre_opcional_del_stash_o_confirmación}}
```

- Suelta o aplica un stash (por defecto es `stash@{0}`) y lo elimina de la lista de stash si su aplicación no causa conflictos:

```
git stash pop {{nombre_opcional_stash}}
```

- Elimina todos los stashes:

```
git stash clear
```

# git status

Muestra los cambios realizados en los archivos del repositorio Git.

Lista los archivos cambiados, añadidos y eliminados comparándolos con el último commit.

Más información: <https://git-scm.com/docs/git-status>.

- Muestra los archivos cambiados que aún no han sido añadidos a un commit:

```
git status
```

- Muestra la salida en formato breve:

```
git status -s
```

- Muestra los archivos rastreados:

```
git status --untracked-files=no
```

- Muestra la salida en formato breve junto a la información del branch:

```
git status --short --branch
```

# git submodule

Inspecciona, actualiza y gestiona los submódulos.

Más información: <https://git-scm.com/docs/git-submodule>.

- Instala los submódulos específicos de un repositorio:

```
git submodule update --init --recursive
```

- Añade un repositorio como un submódulo:

```
git submodule add {{url_del_repositorio}}
```

- Añade un repositorio Git como submódulo en un directorio específico:

```
git submodule add {{url_del_repositorio}} {{ruta/al/directorio}}
```

- Actualiza cada submódulo a su último commit:

```
git submodule foreach git pull
```

# git svn

Operacion bidireccional entre un repositorio Subversión y otro Git.

Más información: <https://git-scm.com/docs/git-svn>.

- Clona un repositorio SVN:

```
git svn clone {{https://ejemplo.com/repositorio_subversion}}  
{{directorio_local}}
```

- Clona un repositorio SVN a partir un número de revisión específico:

```
git svn clone {{-r|--revision}} {{1234}}:HEAD {{https://  
svn.ejemplo.net/subversion/repo}} {{directorio_local}}
```

- Actualiza el clon local a partir del repositorio SVN:

```
git svn rebase
```

- Obtén las actualizaciones del repositorio SVN remoto sin cambiar el HEAD de Git:

```
git svn fetch
```

- Realiza una confirmación en un repositorio SVN:

```
git svn commit
```

# git switch

Alterna entre ramas Git. Requiere una versión 2.23+ de Git.

Vea también **git checkout**.

Más información: <https://git-scm.com/docs/git-switch>.

- Cambia a una rama existente:

```
git switch {{nombre_de_la_rama}}
```

- Crea una nueva rama y se cambia a esta:

```
git switch --create {{nombre_de_la_rama}}
```

- Crea y cambia a una nueva rama basada en una confirmación específica:

```
git switch --create {{nombre_de_la_rama}} {{confirmación}}
```

- Cambia a la rama anterior:

```
git switch -
```

- Cambia a una rama y actualiza todos los submódulos para coincidir:

```
git switch --recurse-submodules {{nombre_de_la_rama}}
```

- Cambia a una rama y automáticamente fusiona la rama actual y cualquier cambio sin confirmación en ella:

```
git switch --merge {{nombre_de_la_rama}}
```

# git tag

Crea, muestra, borra o verifica etiquetas.

Una etiqueta es una referencia estática a una confirmación específica.

Más información: <https://git-scm.com/docs/git-tag>.

- Muestra todas las etiquetas:

```
git tag
```

- Crea una etiqueta con el nombre especificado a partir de la confirmación actual:

```
git tag {{nombre_de_la_etiqueta}}
```

- Crea una etiqueta con el nombre especificado a partir de la confirmación señalada:

```
git tag {{nombre_de_la_etiqueta}} {{confirmación}}
```

- Crea una etiqueta anotada con el mensaje especificado:

```
git tag {{nombre_de_la_etiqueta}} -m  
{{mensaje_de_la_etiqueta}}
```

- Elimina la etiqueta con el nombre especificado:

```
git tag -d {{nombre_de_la_etiqueta}}
```

- Obtén las etiquetas actualizadas de upstreams:

```
git fetch --tags
```

- Muestra todas las etiquetas cuyos ancestros incluyan una confirmación específica:

```
git tag --contains {{confirmación}}
```

# git worktree

Gestiona múltiples árboles de trabajo adjuntos al mismo repositorio.

Más información: <https://git-scm.com/docs/git-worktree>.

- Crea un nuevo directorio con la rama especificada y se cambia a él:

```
git worktree add {{ruta/al/directorio}} {{rama}}
```

- Crea un nuevo directorio con una nueva rama y se cambia a él:

```
git worktree add {{ruta/al/directorio}} -b {{rama_nueva}}
```

- Muestra todos los directorios de trabajo adjuntos a este repositorio:

```
git worktree list
```

- Elimina un árbol de trabajo (después de eliminar el directorio del árbol de trabajo):

```
git worktree prune
```



# git

Sistema de control de versiones distribuido.

Algunos subcomandos como **commit**, **add**, **branch**, **checkout**, **push**, etc., tienen su propia documentación de uso.

Más información: <https://git-scm.com/>.

- Ejecuta un subcomando de Git:

```
git {{subcomando}}
```

- Ejecuta un subcomando de Git en un repositorio en la ruta raíz especificada:

```
git -C {{ruta/al/repositorio}} {{subcomando}}
```

- Ejecuta un subcomando de Git con configuración personalizada:

```
git -c '{{config.clave}}={{valor}}' {{subcomando}}
```

- Muestra ayuda general:

```
git --help
```

- Muestra ayuda sobre un subcomando de Git (p. ej., `clone`, `add`, `push`, `log`, etc.):

```
git help {{subcomando}}
```

- Muestra la versión:

```
git --version
```

# gitleaks

Detecta secretos y claves API filtradas en repositorios Git.

Más información: <https://github.com/gitleaks/gitleaks>.

- Escanea un repositorio remoto:

```
gitleaks detect --repo-url {{https://github.com/  
nombre_usuario/repositorio.git}}
```

- Escanea un directorio local:

```
gitleaks detect --source {{ruta/al/repositorio}}
```

- Crea un archivo JSON con los resultados del análisis:

```
gitleaks detect --source {{ruta/al/repositorio}} --report  
{{ruta/a/informe.json}}
```

- Utiliza un archivo de reglas personalizado:

```
gitleaks detect --source {{ruta/al/repositorio}} --config-  
path {{ruta/a/archivo_de_configuración.toml}}
```

- Inicia la búsqueda a partir de una confirmación específica:

```
gitleaks detect --source {{ruta/al/repositorio}} --log-opts  
{{--since=identificador_confirmación}}
```

- Escanea cambios no confirmados antes de una confirmación:

```
gitleaks protect --staged
```

- Muestra información detallada que indica que partes se identificaron como fugas durante el análisis:

```
gitleaks protect --staged --verbose
```

# gleam

El compilador, la herramienta de compilación, el gestor de paquetes y el formateador de código para Gleam, "un lenguaje amigable para construir sistemas de tipo seguro escalables".

Más información: <https://gleam.run/writing-gleam/command-line-reference/>.

- Crea un nuevo proyecto gleam:

```
gleam new {{nombre_del_proyecto}}
```

- Construye y ejecuta un proyecto gleam:

```
gleam run
```

- Construye el proyecto:

```
gleam build
```

- Ejecuta un proyecto para una plataforma y un tiempo de ejecución específico:

```
gleam run --target {{plataforma}} --runtime  
{{tiempo_de_ejecución}}
```

- Añade una dependencia hexadecimal a tu proyecto:

```
gleam add {{nombre_de_la_dependencia}}
```

- Ejecuta las pruebas del proyecto:

```
gleam test
```

- Formatea el código fuente:

```
gleam format
```

- Comprueba el tipo de proyecto:

```
gleam check
```

# gnmic sub

Este comando es un alias de **gnmic subscribe**.

Más información: <https://gnmic.kmrd.dev/cmd/subscribe>.

- Muestra la documentación del comando original:

```
tldr gnmic subscribe
```

# gocr

Herramienta de reconocimiento óptico de caracteres.

Reconoce caracteres utilizando su motor y solicita al usuario patrones desconocidos para almacenarlos en una base de datos.

Más información: <https://manned.org/gocr.1>.

- Reconoce caracteres en una [i]magen y los escribe en un archiv[o]. Coloca la base de datos en una carpeta existente para que no se omita su uso. [m]odo 130 significa crear, usar y extender la base de datos:

```
gocr -m 130 -p {{ruta/al/directorio_db}} -i {{ruta/a/imagen_entrada.png}} -o {{ruta/al/archivo_salida.txt}}
```

- Reconoce caracteres y asume que todos son números:

```
gocr -m 130 -p {{ruta/al/directorio_db}} -i {{ruta/a/imagen_entrada.png}} -o {{ruta/al/archivo_salida.txt}} -C "{{0..9}}"
```

- Reconoce caracteres con certez[a] del 100% (los caracteres tienen una mayor probabilidad de ser tratados como desconocidos):

```
gocr -m 130 -p {{ruta/al/directorio_db}} -i {{ruta/a/imagen_entrada.png}} -o {{ruta/al/archivo_salida.txt}} -a 100
```

# golangci-lint

Corredor de linters Go paralelizado, inteligente y rápido que se integra con los principales entornos de desarrollo integrado y soporta configuración en YAML.

Más información: <https://golangci-lint.run/welcome/quick-start/>.

- Ejecuta linters en la carpeta actual:

```
golangci-lint run
```

- Lista los linters habilitados y deshabilitados (Nota: los linters deshabilitados se muestran en el último lugar, no los confundas con los habilitados):

```
golangci-lint linters
```

- Habilita un linter específico para esta ejecución:

```
golangci-lint run --enable {{linter}}
```

# google-chrome

Este comando es un alias de **chromium**.

Más información: <https://chrome.google.com>.

- Muestra la documentación del comando original:

```
tldr chromium
```

# gradle

Un sistema de automatización de construcción de código abierto.

Más información: <https://gradle.org>.

- Compila un paquete:

```
gradle build
```

- Excluye la compilación test:

```
gradle build -x {{test}}
```

- Ejecuta en modo sin conexión para evitar que Gradle acceda a la red durante la compilación:

```
gradle build --offline
```

- Limpia el directorio de compilación:

```
gradle clean
```

- Construye un paquete Android (APK) en modo release:

```
gradle assembleRelease
```

- Lista las tareas principales:

```
gradle tasks
```

- Lista todas las tareas:

```
gradle tasks --all
```



# grep

Encuentra patrones en archivos usando expresiones regulares.

Más información: <https://www.gnu.org/software/grep/manual/grep.html>.

- Busca un patrón en un archivo:

```
grep "{{patrón_de_búsqueda}}" {{ruta/al/archivo}}
```

- Busca una cadena de caracteres específica (la cadena no será interpretada como una expresión regular):

```
grep {{-F|--fixed-strings}} "{{cadena_exacta}}" {{ruta/al/archivo}}
```

- Busca un patrón en todos los archivos de forma recursiva en un directorio, mostrando los números de línea de las coincidencias e ignorando los archivos binarios:

```
grep {{-r|--recursive}} {{-n|--line-number}} --binary-files  
{{without-match}} "{{patrón_de_búsqueda}}" {{ruta/al/  
directorio}}
```

- Utiliza expresiones regulares extendidas (los metacaracteres ?, +, {}, () y | no requieren de una barra inversa), sin distinguir entre mayúsculas y minúsculas:

```
grep {{-E|--extended-regexp}} {{-i|--ignore-case}}  
 "{{patrón_de_búsqueda}}" {{ruta/al/archivo}}
```

- Imprime 3 líneas alrededor, antes o después de cada coincidencia:

```
grep --{{context|before-context|after-context}} 3  
 "{{patrón_de_búsqueda}}" {{ruta/al/archivo}}
```

- Imprime con colores el nombre del archivo y el número de línea de cada coincidencia:

```
grep {{-H|--with-filename}} {{-n|--line-number}} --  
color=always "{{patrón_de_búsqueda}}" {{ruta/al/archivo}}
```

- Busca líneas que coincidan con un patrón e imprime solo el texto coincidente:

```
grep {{-o|--only-matching}} "{{patrón_de_búsqueda}}" {{ruta/  
al/archivo}}
```

- Busca líneas en `stdin` que no coincidan con el patrón:

```
cat {{ruta/al/archivo}} | grep {{-v|--invert-match}}  
"{{patrón_de_búsqueda}}"
```

# helm install

Instala un chart de helm.

Más información: [https://helm.sh/docs/intro/using\\_helm/#helm-install-installing-a-package](https://helm.sh/docs/intro/using_helm/#helm-install-installing-a-package).

- Instala un chart de helm:

```
helm install {{nombre}} {{nombre_del_repositorio}}/  
{{nombre_del_chart}}
```

- Instala un chart de helm desde un directorio de chart desempaquetado:

```
helm install {{nombre}} {{ruta/al/directorio_de_origen}}
```

- Instala un chart de helm desde una URL:

```
helm install {{nombre_del_paquete}} {{https://example.com/  
charts/package-1.2.3.tgz}}
```

- Instala un chart de helm y genera un nombre:

```
helm install {{nombre_del_repositorio}}/{{nombre_del_chart}}  
--generate-name
```

- Realiza una simulación:

```
helm install {{nombre}} {{nombre_del_repositorio}}/  
{{nombre_del_chart}} --dry-run
```

- Instala un chart de helm con valores personalizados:

```
helm install {{nombre}} {{nombre_del_repositorio}}/  
{{nombre_del_chart}} --set {{parámetro1}}={{valor1}},  
{{parámetro2}}={{valor2}}
```

- Instala un chart de helm pasando un archivo de valores personalizados:

```
helm install {{nombre}} {{nombre_del_repositorio}}/  
{{nombre_del_chart}} --values {{ruta/a/valores.yaml}}
```

# helm

Helm es un gestor de paquetes para Kubernetes.

Algunos subcomandos como **install** tiene su propia documentación de uso.

Más información: <https://helm.sh/>.

- Crea un chart de helm:

```
helm create {{nombre_del_chart}}
```

- Añade un nuevo repositorio de helm:

```
helm repo add {{nombre_del_repositorio}}
```

- Lista de repositorios de helm:

```
helm repo list
```

- Actualiza los repositorios de helm:

```
helm repo update
```

- Elimina un repositorio de helm:

```
helm repo remove {{nombre_del_repositorio}}
```

- Instala un chart de helm:

```
helm install {{nombre}} {{nombre_del_repositorio}}/  
{{nombre_del_chart}}
```

- Descarga un chart de helm como un archivo tar:

```
helm get {{nombre_del_lanzamiento_del_chart}}
```

- Actualiza las dependencias de helm:

```
helm dependency update
```

# hexyl

Un simple visor hexadecimal para la terminal. Utiliza salida coloreada para distinguir diferentes categorías de bytes.

Más información: <https://github.com/sharkdp/hexyl>.

- Imprime la representación hexadecimal de un archivo:

```
hexyl {{ruta/al/archivo}}
```

- Imprime la representación hexadecimal de los primeros `n` bytes de un archivo:

```
hexyl -n {{n}} {{ruta/al/archivo}}
```

- Imprime los bytes 512 a 1024 de un archivo:

```
hexyl -r {{512}}:{{1024}} {{ruta/al/archivo}}
```

- Imprime 512 bytes empezando por el byte 1024:

```
hexyl -r {{1024}}:+{{512}} {{ruta/al/archivo}}
```

# history expansion

Reutiliza y expande el historial del shell en **sh**, Bash, Zsh, **rbash** y **ksh**.

Más información: [https://www.gnu.org/software/bash/manual/html\\_node/History-Interaction](https://www.gnu.org/software/bash/manual/html_node/History-Interaction).

- Ejecuta el comando anterior como root (**!!** se sustituye por el comando anterior):

```
sudo !!
```

- Ejecuta un comando con el último argumento del comando anterior:

```
{{comando}} !$
```

- Ejecuta un comando con el primer argumento del comando anterior:

```
{{comando}} !^
```

- Ejecuta el **n** comando del historial:

```
!{{n}}
```

- Ejecuta el comando **n** líneas atrás en el historial:

```
!-{{n}}
```

- Ejecuta el comando más reciente que contenga **cadena**:

```
!?{{{cadena}}}
```

- Ejecuta el comando anterior, sustituyendo "cadena1" por "cadena2":

```
^{{{cadena1}}}^{{{cadena2}}}^
```

- Realiza una expansión del historial, pero imprimiendo el comando que se ejecutaría en lugar de ejecutarlo realmente:

```
{{!-n}}:p
```

# history

Historial de la línea de comandos.

Más información: [https://www.gnu.org/software/bash/manual/html\\_node/Bash-History-Builtins.html](https://www.gnu.org/software/bash/manual/html_node/Bash-History-Builtins.html).

- Muestra el historial de comandos junto a su número de línea:

```
history
```

- Muestra los últimos 20 comandos:

```
history {{20}}
```

- Limpia el historial de comandos (solo para la shell actual):

```
history -c
```

- Sobrescribe el archivo histórico con el historial de la shell actual (comúnmente se combina con `history -c` para limpiar el historial):

```
history -w
```

- Borra la entrada del historial en el índice especificado:

```
history -d {{indice}}
```

# hledger balance

Un informe "sumatorio" flexible y de propósito general que muestra cuentas con algún tipo de dato numérico.

Puede tratarse de cambios de saldo por periodo, saldos finales, rendimiento presupuestario, plusvalías latentes, etc.

Más información: <https://hledger.org/hledger.html#balance>.

- Muestra el cambio de saldo en todas las cuentas de todas las contabilizaciones a lo largo de todo el tiempo:

```
hledger balance
```

- Muestra el cambio de saldo en las cuentas denominadas *\*gastos\**, como un árbol, resumiendo solo los dos niveles superiores:

```
hledger balance {{gastos}} --tree --depth {{2}}
```

- Muestra los gastos de cada mes, y sus totales y medias, ordenados por total; y sus objetivos presupuestarios mensuales:

```
hledger balance {{gastos}} --monthly --row-total --average --sort-amount --budget
```

- Similar a la anterior, de forma más corta, comparando las cuentas por tipo de Gastos, como un árbol de dos niveles sin aplastar las cuentas aburridas:

```
hledger bal type:{{X}} -MTAS --budget -t -{{2}} --no-elide
```

- Muestra saldos finales (incluidos los de contabilizaciones anteriores a la fecha de inicio), trimestrales en 2024, en cuentas denominadas *\*activos\** o *\*pasivos\**:

```
hledger balance --historical --period '{{trimestral en 2024}}' {{activos}} {{pasivos}}
```

- Similar al anterior, de un modo más breve; también muestra saldos en cero, ordena por total y resume a tres niveles:

```
hledger bal -HQ date:{{2024}} type:{{AL}} -ES -{{3}}
```

- Muestra el valor de mercado de los activos de inversión en moneda base al final de cada trimestre:



```
hledger bal -HVQ {{activos:inversiones}}
```

- Muestra las ganancias/pérdidas de capital no realizadas por cambios en el precio de mercado en cada trimestre, para activos de inversión que no sean criptomonedas:

```
hledger bal --gain -Q {{activos:inversiones}} not:{{criptomoneda}}
```

# htop

Muestra información dinámica en tiempo real sobre los procesos ejecutándose.  
Una versión mejorada de **top**.

Más información: <https://htop.dev/>.

- Inicia htop:

```
htop
```

- Inicia htop mostrando solo procesos pertenecientes a un usuario dado:

```
htop --user {{nombre_de_usuario}}
```

- Ordena procesos por un específico `elemento_de_ordenamiento` (use `htop --sort help` para opciones disponibles):

```
htop --sort {{elemento_de_ordenamiento}}
```

- Muestra comandos interactivos mientras corre htop:

```
?
```

- Muestra la ayuda:

```
htop --help
```

# https

Este comando es un alias de **http**.

- Consulte la documentación del comando original:

```
tldr http
```

# huggingface-cli

Interactúa con Hugging Face Hub.

Inicia sesión, gestiona la caché local, carga o descarga archivos.

Más información: [https://huggingface.co/docs/huggingface\\_hub/guides/cli](https://huggingface.co/docs/huggingface_hub/guides/cli).

- Inicia sesión en Hugging Face Hub:

```
huggingface-cli login
```

- Muestra el nombre del usuario conectado:

```
huggingface-cli whoami
```

- Cierra sesión:

```
huggingface-cli logout
```

- Genera información sobre el entorno:

```
huggingface-cli env
```

- Descarga archivos de un repositorio e imprime la ruta (omite los nombres de archivo para descargar todo el repositorio):

```
huggingface-cli download --repo-type {{repo_type}}  
{{repo_id}} {{nombre_archivo1 nombre_archivo2 ...}}
```

- Sube una carpeta entera o un archivo a Hugging Face:

```
huggingface-cli upload --repo-type {{repo_type}} {{repo_id}}  
{{ruta/al/  
archivo_de_repositorio_o_directorio_de_repositorio}} {{ruta/  
al/archivo_de_repositorio_o_directorio}}
```

- Escanea la caché para ver los repositorios descargados y su uso de disco:

```
huggingface-cli scan-cache
```

- Elimina la caché de forma interactiva:

```
huggingface-cli delete-cache
```

# hugo server

Construye y publica un sitio con el servidor web integrado de Hugo.

Más información: [https://gohugo.io/commands/hugo\\_server/](https://gohugo.io/commands/hugo_server/).

- Construye y publica un sitio:

```
hugo server
```

- Construye y publica un sitio en un número de puerto especificado:

```
hugo server --port {{número_de_puerto}}
```

- Construye y publica un sitio mientras se minifican los formatos de salida soportados (HTML, XML, etc.):

```
hugo server --minify
```

- Muestra la ayuda:

```
hugo server --help
```

# hx

Este comando es un alias de **helix**.

- Muestra la documentación del comando original:

```
tldr helix
```

# id

Muestra la identidad actual del usuario y del grupo.

Más información: <https://www.gnu.org/software/coreutils/id>.

- Muestra la ID del usuario actual (UID), la ID del grupo (GID) y los grupos a los que pertenece:

```
id
```

- Muestra la identidad del usuario actual:

```
id -un
```

- Muestra la identidad del usuario actual como un número:

```
id -u
```

- Muestra la identidad del grupo primario actual:

```
id -gn
```

- Muestra la identidad del grupo primario actual como un número:

```
id -g
```

- Muestra el ID (UID) de un usuario arbitrario, el ID de grupo (GID) y los grupos a los que pertenece:

```
id {{nombre_de_usuario}}
```

# immich-cli

Immich tiene una interfaz de línea de comandos (CLI) que le permite realizar ciertas acciones desde la línea de comandos.

Vea también: **immich-go**.

Más información: <https://immich.app/docs/features/command-line-interface/>.

- Autentica en el servidor de Immich:

```
immich login {{url_del_servidor/api}} {{clave_del_servidor}}
```

- Sube unas imágenes:

```
immich upload {{archivo1.jpg archivo2.jpg}}
```

- Sube un directorio y sus subdirectorios:

```
immich upload --recursive {{ruta/al/directorio}}
```

- Crea un álbum basado en un directorio:

```
immich upload --album-name "{{Vacaciones de verano}}" --recursive {{ruta/al/directorio}}
```

- Omite recursos que coincidan con un patrón global:

```
immich upload --ignore {{**/Raw/** **/*.tif}} --recursive {{directorio/}}
```

- Incluye archivos ocultos:

```
immich upload --include-hidden --recursive {{ruta/al/directorio}}
```



# immich-go

Immich-Go es una herramienta abierta diseñada para subir grandes colecciones de fotos a tu servidor Immich autoalojado.

Véase también: **immich-cli**.

Más información: <https://github.com/simulot/immich-go>.

- Sube un archivo takeout de Google al servidor Immich:

```
immich-go -server={{url_del_servidor}} -  
key={{clave_del_servidor}} upload {{ruta/a/  
archivo_takeout.zip}}
```

- Importa fotos capturadas en junio del 2019, mientras se generan los álbumes automáticamente:

```
immich-go -server={{url_del_servidor}} -  
key={{clave_del_servidor}} upload -create-albums -google-  
photos -date={{2019-06}} {{ruta/a/archivo_takeout.zip}}
```

- Sube un archivo usando servidor y clave de un archivo de configuración:

```
immich-go -use-configuration={{~/immich-go/immich-go.json}}  
upload {{ruta/a/archivo_takeout.zip}}
```

- Examina el contenido del servidor Immich, elimina las imágenes de menor calidad y preserva álbumes:

```
immich-go -server={{url_del_servidor}} -  
key={{clave_del_servidor}} duplicate -yes
```

- Elimina todos los álbumes creados con el patrón "YYYY-MM-DD":

```
immich-go -server={{url_del_servidor}} -  
key={{clave_del_servidor}} tool album delete {{\d{4}-\d{2}-  
\d{2}}}
```

# ipscan

Un rápido escáner de red diseñado para ser simple de usar.

También conocido como Angry IP Scanner.

Más información: <https://angryip.org/>.

- Escanea una dirección IP específica:

```
ipscan {{192.168.0.1}}
```

- Escanea un rango de direcciones IP:

```
ipscan {{192.168.0.1-254}}
```

- Escanea un rango de direcciones IP y guardar los resultados en un archivo:

```
ipscan {{192.168.0.1-254}} -o {{ruta/a/salida.txt}}
```

- Escanea IPs con un conjunto específico de puertos:

```
ipscan {{192.168.0.1-254}} -p {{80,443,22}}
```

- Escanea con un retardo entre peticiones para evitar la congestión de la red:

```
ipscan {{192.168.0.1-254}} -d {{200}}
```

- Muestra ayuda:

```
ipscan --help
```

# ispell

Corrección ortográfica interactiva.

Más información: <https://www.cs.hmc.edu/~geoff/ispell-man.html>.

- Inicia una sesión interactiva:

```
ispell
```

- Comprueba si hay erratas en el archivo especificado y aplica sugerencias de forma interactiva:

```
ispell {{ruta/al/archivo}}
```

- Muestra la versión:

```
ispell -v
```

# jbang

Crea, edita y ejecuta fácilmente programas en Java autocontenidos de sólo código fuente.

Ver también: **java**.

Más información: <https://www.jbang.dev/documentation/guide/latest/cli/jbang.html>.

- Inicializa una clase en Java simple:

```
jbang init {{ruta/al/archivo.java}}
```

- Inicializa una clase en Java (útil para scripts):

```
jbang init --template={{cli}} {{ruta/al/archivo.java}}
```

- Utiliza `jshell` para explorar y utilizar un script y cualquier dependencia en un editor REPL:

```
jbang run --interactive
```

- Configura un proyecto temporal para editar un script en un entorno de desarrollo integrado:

```
jbang edit --open={{codium|code|eclipse|idea|netbeans|gitpod}} {{ruta/al/script.java}}
```

- Ejecuta un fragmento de código en Java (Java 9 y posteriores):

```
{{echo 'Files.list(Paths.get("/  
etc')).forEach(System.out::println);'}} | jbang -
```

- Ejecuta aplicación de línea de comandos:

```
jbang {{ruta/al/archivo.java}} {{comando}} {{arg1 arg2 ...}}
```

- Instala un script en un directorio en el valor de la variable de entorno `PATH` del usuario actual:

```
jbang app install --name {{nombre_del_comando}} {{ruta/al/  
script.java}}
```

- Instala una versión específica del JDK para utilizarla con `jbang`:

```
jbang jdk install {{versión}}
```

# kafkacat

Este comando es un alias de **kcat**.

- Muestra la documentación del comando original:

```
tldr kcat
```

# keybase

Directorio de claves que conecta identidades en redes sociales a claves encriptadas de una manera públicamente auditable.

Más información: [https://keybase.io/docs/command\\_line](https://keybase.io/docs/command_line).

- Sigue a otro usuario:

```
keybase follow {{nombre_de_usuario}}
```

- Añade una nueva prueba:

```
keybase prove {{servicio}}  
{{nombre_de_usuario_en_el_servicio}}
```

- Firma un archivo:

```
keybase sign --infile {{archivo_de_entrada}} --outfile  
{{archivo_de_salida}}
```

- Verifica un archivo firmado:

```
keybase verify --infile {{archivo_de_entrada}} --outfile  
{{archivo_de_salida}}
```

- Encripta un archivo:

```
keybase encrypt --infile {{archivo_de_entrada}} --outfile  
{{archivo_de_salida}} {{receptor}}
```

- Desencripta un archivo:

```
keybase decrypt --infile {{archivo_de_entrada}} --outfile  
{{archivo_de_salida}}
```

- Revoca el dispositivo actual, se desconecta y borra los datos locales:

```
keybase deprovision
```

# kill

Envía una señal a un proceso, usualmente relacionado a detener el proceso.

Kill envía una señal para terminar uno o un grupo de procesos.

Más información: <https://manned.org/kill.1posix>.

- Termina un programa usando la señal SIGTERM (terminar) por defecto:

```
kill {{identificador_del_proceso}}
```

- Lista todas las señales disponibles (para ser utilizadas sin el prefijo SIG):

```
kill -l
```

- Termina una tarea en segundo plano:

```
kill %{{identificador_de_tarea}}
```

- Termina un programa usando la señal SIGHUP (hang up/colgar). Muchos programas residentes se recargarán en lugar de terminar:

```
kill -{{1|HUP}} {{identificador_del_proceso}}
```

- Termina un programa usando la señal SIGINT (interrumpir). Esto es normalmente iniciado por el usuario presionando Ctrl + C:

```
kill -{{2|INT}} {{identificador_del_proceso}}
```

- Señala al sistema operativo para terminar inmediatamente un programa (el cual no tiene oportunidad de capturar la señal):

```
kill -{{9|KILL}} {{identificador_del_proceso}}
```

- Señala al sistema operativo para pausar un programa hasta que la señal SIGCONT ("continuar") es recibida:

```
kill -{{17|STOP}} {{identificador_del_proceso}}
```

- Envía una señal SIGUSR1 a todos los procesos con un GID (id de grupo) dado:

```
kill -{{SIGUSR1}} -{{identificador_de_grupo}}
```

# knotc

Controla el servidor DNS knot.

Más información: [https://www.knot-dns.cz/docs/latest/html/man\\_knotc.html](https://www.knot-dns.cz/docs/latest/html/man_knotc.html).

- Comienza a editar una zona:

```
knotc zone-begin {{zona}}
```

- Establece un registro A con TTL de 3600:

```
knotc zone-set {{zona}} {{subdominio}} 3600 A  
{{dirección_ip}}
```

- Finaliza la edición de la zona:

```
knotc zone-commit {{zona}}
```

- Obtén los datos de la zona actual:

```
knotc zone-read {{zona}}
```

- Obtén la configuración actual del servidor:

```
knotc conf-read server
```



# kopia

Herramienta de copia de seguridad de código abierto, rápida y segura.

Soporta encriptación, compresión, deduplicación e instantáneas incrementales.

Más información: <https://kopia.io/docs/reference/command-line/>.

- Crea un repositorio en el sistema de archivos local:

```
kopia repository create filesystem --path {{ruta/al/
repositorio_local}}
```

- Crea un repositorio en Amazon S3:

```
kopia repository create s3 --bucket {{nombre_del_bucket}} --
access-key {{identificador_de_clave_de_acceso_AWS}} --secret-
access-key {{clave_de_acceso_secreta_AWS}}
```

- Conecta a un repositorio:

```
kopia repository connect {{tipo_de_repositorio}} --path
{{ruta/al/repositorio}}
```

- Crea una instantánea de un directorio:

```
kopia snapshot create {{ruta/al/directorio}}
```

- Lista instantáneas:

```
kopia snapshot list
```

- Restaura una instantánea en un directorio específico:

```
kopia snapshot restore {{identificador_de_instantánea}}
{{ruta/al/directorio_objetivo}}
```

- Crea una nueva política:

```
kopia policy set --global --keep-latest
{{número_de_instantáneas_a_mantener}} --compression
{{algoritmo_de_compresión}}
```

- Excluye un archivo o directorio específico de las copias de seguridad:

```
kopia policy set --global --add-ignore {{ruta/al/
archivo_o_directorio}}
```

# krunvm

Utilidad basada en CLI para crear micro máquinas virtuales utilizando imágenes OCI.

Más información: <https://github.com/containers/krunvm>.

- Crea una micro máquina virtual basada en Fedora:

```
krunvm create {{docker.io/fedora}} --cpus {{número_de_vcpus}}  
--mem {{memoria_en_megabytes}} --name "{{nombre}}"
```

- Inicia una imagen específica:

```
krunvm start "{{nombre}}"
```

- Lista las imágenes existentes:

```
krunvm list
```

- Cambia una imagen específica:

```
krunvm changevm --cpus {{número_de_vcpus}} --mem  
{{memoria_en_megabytes}} --name "{{nuevo_nombre}}"  
"{{nombre}}"
```

- Borra una imagen específica:

```
krunvm delete "{{nombre}}"
```

# kubectl logs

Muestra los registros de los contenedores de un pod.

Más información: <https://kubernetes.io/docs/reference/generated/kubectl/kubectl-commands#logs>.

- Muestra los registros de un pod de un contenedor:

```
kubectl logs {{nombre_del_pod}}
```

- Muestra los registros de un contenedor especificado en un pod:

```
kubectl logs --container {{nombre_del_contenedor}}  
{{nombre_del_contenedor}}
```

- Muestra los registros de todos los contenedores de un pod:

```
kubectl logs --all-containers={{true}}  
{{nombre_del_contenedor}}
```

- Transmite los registros del pod:

```
kubectl logs --follow {{nombre_del_pod}}
```

- Muestra los registros de pods más recientes dado un tiempo relativo como 10s, 5m o 1h:

```
kubectl logs --since={{tiempo_relativo}} {{nombre_del_pod}}
```

- Muestra los 10 registros más recientes de un pod:

```
kubectl logs --tail={{10}} {{nombre_del_pod}}
```

- Muestra todos los registros de un pod para un despliegue determinado:

```
kubectl logs deployment/{{nombre_del_despliegue}}
```

# llvm-ar

Este comando es un alias de **ar**.

- Muestra la documentación del comando original:

```
tldr ar
```

# llvm-g++

Este comando es un alias de **clang++**.

- Muestra la documentación del comando original:

```
tldr clang++
```

# llvm-gcc

Este comando es un alias de **clang**.

- Muestra la documentación del comando original:

```
tldr clang
```

# llvm-nm

Este comando es un alias de **nm**.

- Muestra la documentación del comando original:

```
tldr nm
```

# llvm-objdump

Este comando es un alias de **objdump**.

- Muestra la documentación del comando original:

```
tldr objdump
```



# llvm-strings

Este comando es un alias de **strings**.

- Muestra la documentación del comando original:

```
tldr strings
```

# ln

Crea enlaces a archivos y directorios.

Más información: <https://www.gnu.org/software/coreutils/ln>.

- Crea un enlace simbólico a un archivo o directorio:

```
ln -s {{ruta/al/archivo_o_directorio}} {{ruta/al/enlace_simbólico}}
```

- Sobrescribe un enlace simbólico existente para que apunte a un archivo distinto:

```
ln -sf {{ruta/al/nuevo_archivo}} {{ruta/al/enlace_simbólico}}
```

- Crea un enlace duro a un archivo:

```
ln {{ruta/al/archivo}} {{ruta/al/enlace_duro}}
```

# ls

Lista los contenidos de directorios.

Más información: <https://www.gnu.org/software/coreutils/ls>.

- Lista un archivo por línea:

```
ls -l
```

- Lista todos los archivos, incluyendo archivos ocultos:

```
ls -a
```

- Lista todos los archivos, añadiendo / al final de los nombres de directorios:

```
ls -F
```

- Lista todos los archivos en formato largo (permisos, propietarios, tamaño y fecha de última modificación):

```
ls -la
```

- Lista en formato largo y tamaño legible por humanos (i.e., KiB, MiB, GiB, etc.):

```
ls -lh
```

- Lista recursivamente en formato largo y ordena los tamaños de mayor a menor:

```
ls -lSR
```

- Lista todos los archivos en formato largo y ordenados por fecha de modificación (archivos más viejos en primer lugar):

```
ls -ltr
```

- Lista solamente directorios:

```
ls -d */
```

# lstopo

Muestra la topología de hardware del sistema.

Más información: <https://manned.org/lstopo>.

- Muestra la topología resumida del sistema en una ventana gráfica (o imprime en consola si no se dispone de una sesión gráfica):

```
lstopo
```

- Muestra la topología completa del sistema sin resúmenes:

```
lstopo --no-factorize
```

- Muestra la topología resumida del sistema sólo con índices físicos (es decir, tal y como la ve el sistema operativo):

```
lstopo --physical
```

- Escribe la topología completa del sistema en un archivo con el formato especificado:

```
lstopo --no-factorize --output-format {{console|ascii|tex|  
fig|svg|pdf|ps|png|xml}} {{ruta/al/archivo}}
```

- Muestra datos en monocromo o escala de grises:

```
lstopo --palette {{none|grey}}
```

# lzcat

Este comando es un alias de **xz**.

Más información: <https://manned.org/lzcat>.

- Muestra la documentación del comando original:

```
tldr xz
```

# lzma

Este comando es un alias de **xz**.

Más información: <https://manned.org/lzma>.

- Muestra la documentación del comando original:

```
tldr xz
```

# maza

Bloqueador en local de anuncios. Como Pi-hole pero local y usando el sistema operativo.

Más información: <https://github.com/tanrax/maza-ad-blocking>.

- Actualiza la base de datos de Maza:

```
maza update
```

- Inicia Maza:

```
sudo maza start
```

- Para Maza:

```
sudo maza stop
```

- Muestra el estado de Maza:

```
maza status
```

# medusa

Un forzador bruto de inicio de sesión modular y paralelo para una variedad de protocolos.

Más información: <https://jmk-foofus.github.io/medusa/medusa.html>.

- Lista todos los módulos instalados:

```
medusa -d
```

- Muestra ejemplo de uso de un módulo específico (usa `medusa -d` para listar todos los módulos instalados):

```
medusa -M {{ssh|http|web-form|postgres|ftp|mysql|...}} -q
```

- Ejecuta fuerza bruta contra un servidor FTP utilizando un fichero que contiene nombres de usuario y un fichero que contiene contraseñas:

```
medusa -M ftp -h host -U {{ruta/a/  
archivo_de_nombre_de_usuario}} -P {{ruta/a/  
archivo_contraseña}}
```

- Ejecuta un intento de inicio de sesión contra un servidor HTTP utilizando el nombre de usuario, la contraseña y el agente de usuario especificados:

```
medusa -M HTTP -h host -u {{nombre_usuario}} -p  
{{contraseña}} -m USER-AGENT:"{{Agente}}"
```

- Ejecuta una fuerza bruta contra un servidor MySQL utilizando un fichero que contenga nombres de usuario y un hash:

```
medusa -M mysql -h host -U {{ruta/a/  
archivo_de_nombre_de_usuario}} -p {{hash}} -m PASS:HASH
```

- Ejecuta una fuerza bruta contra una lista de servidores SMB utilizando un nombre de usuario y un archivo pwdump:

```
medusa -M smbnt -H {{ruta/a/archivo_de_hosts}} -C {{ruta/a/  
archivo_pwdump}} -u {{nombre_usuario}} -m PASS:HASH
```



# mid3v2

Edita etiquetas de audio.

Vea también: **id3v2**.

Más información: <https://mutagen.readthedocs.io/en/latest/man/mid3v2.html>.

- Lista de todos los marcos ID3v2.3 o ID3v2.4 admitidos y sus significados:

```
id3v2 --list-frames {{ruta/al/archivo1.mp3 ruta/al/
archivo2.mp3 ...}}
```

- Lista de todos los géneros numéricos ID3v1 admitidos:

```
id3v2 --list-genres {{ruta/al/archivo1.mp3 ruta/al/
archivo2.mp3 ...}}
```

- Lista todas las etiquetas en archivos específicos:

```
id3v2 --list {{ruta/al/archivo1.mp3 ruta/al/archivo2.mp3
...}}
```

- Establece información específica sobre artistas, álbumes o canciones:

```
id3v2 {{--artist|--album|--song}}={{string}} {{ruta/al/
archivo1.mp3 ruta/al/archivo2.mp3 ...}}
```

- Establece información específica de la imagen:

```
id3v2 --picture={{filename:description:image_type:mime_type}}
{{ruta/al/archivo1.mp3 ruta/al/archivo2.mp3 ...}}
```

- Establece información específica del año:

```
id3v2 --year={{YYYY}} {{ruta/al/archivo1.mp3 ruta/al/
archivo2.mp3 ...}}
```

- Establece información de fecha específica:

```
id3v2 --date={{YYYY-MM-DD}} {{ruta/al/archivo1.mp3 ruta/al/
archivo2.mp3 ...}}
```

# mise

Gestiona versiones de diferentes paquetes.

Más información: <https://mise.jdx.dev>.

- Lista todos los complementos disponibles:

```
mise plugins list-all
```

- Instala un complemento:

```
mise plugins add {{nombre}}
```

- Lista las versiones disponibles para instalación:

```
mise ls-remote {{nombre}}
```

- Instala una versión específica de un paquete:

```
mise install {{nombre}}@{{versión}}
```

- Establece una versión global de un paquete:

```
mise use --global {{nombre}}@{{versión}}
```

- Establece una versión local de un paquete:

```
mise use {{nombre}}@{{versión}}
```

- Establece una variable de entorno en la configuración:

```
mise set {{variable}}={{valor}}
```

# mitmproxy

Un proxy HTTP interactivo man-in-the-middle.

Vea también: **mitmweb** y **mitmdump**.

Más información: <https://docs.mitmproxy.org/stable/>.

- Inicia **mitmproxy** con la configuración por defecto (escuchará en el puerto 8080):

```
mitmproxy
```

- Inicia **mitmproxy** con una dirección y puerto personalizados:

```
mitmproxy --listen-host {{dirección_ip}} {-p|--listen-port} {{puerto}}
```

- Inicia **mitmproxy** utilizando un script para procesar el tráfico:

```
mitmproxy {-s|--scripts} {{ruta/a/script.py}}
```

- Exporta los registros con las claves maestras SSL/TLS a programas externos (wireshark, etc.):

```
SSLKEYLOGFILE="{{ruta/a/archivo}}" mitmproxy
```

- Especifica el modo de funcionamiento del servidor proxy (**regular** es el predeterminado):

```
mitmproxy {-m|--mode} {{regular|transparent|socks5|...}}
```

- Configura el diseño de la consola:

```
mitmproxy --console-layout {{horizontal|single|vertical}}
```

# mkdir

Crea un directorio.

Más información: <https://www.gnu.org/software/coreutils/mkdir>.

- Crea un directorio en el directorio actual o en una ruta dada:

```
mkdir {{directorio}}
```

- Crea directorios recursivamente (útil para crear directorios anidados):

```
mkdir {{-p|--parents}} {{ruta/al/directorio}}
```

# mkfifo

Crea FIFOs (llamadas pipes).

Más información: <https://www.gnu.org/software/coreutils/mkfifo>.

- Crea un pipe con nombre en una ruta dada:

```
mkfifo {{ruta/a/pipe}}
```

- Envía datos a través de un pipe con nombre y envía el comando al fondo:

```
echo {"Hola Mundo"} > {{ruta/a/pipe}} &
```

- Recibe datos a través de un pipe con nombre:

```
cat {{ruta/a/pipe}}
```

# more

Interactivamente muestra el contenido un archivo, permitiendo desplazamiento y búsqueda.

Vea también: **less**.

Más información: <https://manned.org/more.1p>.

- Abre un archivo:

```
more {{ruta/al/archivo}}
```

- Abre un archivo y muestra su contenido desde una línea específica:

```
more +{{número_de_línea}} {{ruta/al/archivo}}
```

- Avanza a la siguiente página:

```
<Espacio>
```

- Busca una cadena de caracteres (presione **n** para ir a la siguiente coincidencia):

```
/{{cadena}}
```

- Sal del programa:

```
q
```

- Muestra ayuda sobre comandos interactivos:

```
h
```

# mscore

Este comando es un alias de **musescore**.

Más información: <https://musescore.org/handbook/command-line-options>.

- Muestra la documentación del comando original:

```
tldr musescore
```

# msfconsole

Consola para el Metasploit Framework.

Más información: <https://docs.rapid7.com/metasploit/msf-overview>.

- Inicia la consola:

```
msfconsole
```

- Lanza la consola [q]uietamente sin ningún mensaje:

```
msfconsole --quiet
```

- No habilita el soporte de bases de datos:

```
msfconsole --no-database
```

- Ejecuta los comandos de la consola (Nota: utiliza ; para pasar varios comandos):

```
msfconsole --execute-command "{{use auxiliary/server/capture/ftp; set SRVHOST 0.0.0.0; set SRVPORT 21; run}}"
```

- Muestra [v]ersión:

```
msfconsole --version
```



# mv

Mueve o renombra archivos y directorios.

Más información: <https://www.gnu.org/software/coreutils/mv>.

- Cambia el nombre de un archivo o directorio cuando el destino no es un directorio existente:

```
mv {{ruta/a/origen}} {{ruta/a/destino}}
```

- Mueve un archivo o directorio a un directorio existente:

```
mv {{ruta/a/origen}} {{ruta/a/directorio_existente}}
```

- Mueve varios archivos a un directorio existente, manteniendo los nombres de archivo sin cambios:

```
mv {{ruta/a/origen1 ruta/a/origen2 ...}} {{ruta/a/directorio_existente}}
```

- No pedir confirmación ([f]) antes de sobrescribir los archivos existentes:

```
mv --force {{ruta/a/origen}} {{ruta/a/destino}}
```

- Pedir confirmación [i]nteractivamente antes de sobrescribir archivos existentes, independientemente de los permisos de los archivos:

```
mv --interactive {{ruta/a/origen}} {{ruta/a/destino}}
```

- No sobrescribir ([n]) los archivos existentes en el destino:

```
mv --no-clobber {{ruta/a/origen}} {{ruta/a/destino}}
```

- Mueve archivos en modo [v]erbose, mostrando los archivos después de moverlos:

```
mv --verbose {{ruta/a/origen}} {{ruta/a/destino}}
```

- Especifica el directorio de des[t]ino para poder utilizar herramientas externas para recopilar archivos movibles:

```
{{find /var/log -type f -name '*.log' -print0}} | {{xargs -0}} mv --target-directory {{ruta/a/directorio_destino}}
```

# mypy

Verificación de código Python.

Más información: [https://mypy.readthedocs.io/en/stable/running\\_mypy.html](https://mypy.readthedocs.io/en/stable/running_mypy.html).

- Comprueba un archivo específico:

```
mypy {{ruta/al/archivo.py}}
```

- Comprueba un [m]ódulo específico:

```
mypy -m {{nombre_del_módulo}}
```

- Comprueba un [p]aquete específico:

```
mypy -p {{nombre_del_paquete}}
```

- Comprueba una cadena de código:

```
mypy -c "{{código}}"
```

- Ignora importaciones faltantes:

```
mypy --ignore-missing-imports {{ruta/  
al_archivo_o_directorio}}
```

- Muestra mensajes de error detallados:

```
mypy --show-traceback {{ruta/al/archivo_o_directorio}}
```

- Especifica un archivo de configuración personalizado:

```
mypy --config-file {{ruta/al/archivo_de_configuración}}
```

- Muestra ayuda:

```
mypy -h
```

# mysql

Herramienta de línea de comandos para gestionar bases de datos MySQL.

Más información: <https://www.mysql.com/>.

- Conecta a una base de datos:

```
mysql {{nombre_base_de_datos}}
```

- Conecta a una base de datos con el usuario `usuario` y se le pedirá la contraseña:

```
mysql -u {{usuario}} --password {{nombre_base_de_datos}}
```

- Conecta a una base de datos en otra máquina:

```
mysql -h {{maquina_remota}} {{nombre_base_de_datos}}
```

- Conecta a una base de datos a través de un socket unix:

```
mysql --socket {{ruta/al/socket.sock}}
```

- Ejecuta comandos SQL contenidos en un script:

```
mysql -e "source {{archivo.sql}}" {{nombre_base_de_datos}}
```

- Restaura una base de datos a partir de una copia de seguridad creada con `mysqldump` (y se le pedirá la contraseña al usuario):

```
mysql --user {{usuario}} --password {{nombre_base_de_datos}}  
< {{ruta/al/backup.sql}}
```

- Restaura todas las bases de datos en una copia de seguridad (y se le pedirá la contraseña al usuario):

```
mysql --user {{usuario}} --password < {{ruta/al/backup.sql}}
```

# mysqldump

Crea una copia de seguridad de bases de datos MySQL.

Vea también **mysql** para restaurar bases de datos.

Más información: <https://dev.mysql.com/doc/refman/en/mysqldump.html>.

- Crea un backup (se le pedirá la contraseña al usuario):

```
mysqldump --user {{usuario}} --password  
{{nombre_base_de_datos}} -r {{ruta/al/archivo.sql}}
```

- Crea un backup de todas las bases de datos y redirige la salida a un archivo (se le pedirá la contraseña al usuario):

```
mysqldump --user {{usuario}} --password --all-databases >  
{{ruta/al/archivo.sql}}
```

- Crea un backup de una única tabla de una base de datos (se le pedirá la contraseña al usuario):

```
mysqldump --user {{usuario}} --password  
{{nombre_base_de_datos}} {{nombre_tabla}} -r {{ruta/al/  
archivo.sql}}
```

# naabu

Un rápido escáner de puertos escrito en Go con un enfoque en la fiabilidad y la simplicidad.

Nota: Algunas características solo se activan cuando **naabu** se ejecuta con privilegios del superusuario, como el escaneo SYN.

Más información: <https://github.com/projectdiscovery/naabu>.

- Ejecuta un escaneo SYN contra los puertos predeterminados (top 100) del host remoto:

```
sudo naabu -host {{host}}
```

- Muestra las interfaces de red disponibles y la dirección IP pública del host local:

```
naabu -interface-list
```

- Escanea todos los puertos del host remoto (escaneo CONNECT sin sudo):

```
naabu -p - -host {{host}}
```

- Escanea los top 1000 puertos del host remoto:

```
naabu -top-ports 1000 -host {{host}}
```

- Escanea los puertos TCP 80, 443 y UDP 53 del host remoto:

```
naabu -p 80,443,u:53 -host {{host}}
```

- Muestra el tipo de CDN que utiliza el host remoto, si existe:

```
naabu -p 80,443 -cdn -host {{host}}
```

# nano

Editor sencillo y fácil de usar. Un clon libre y mejorado de Pico.

Más información: <https://nano-editor.org>.

- Abre un nuevo archivo en nano:

```
nano
```

- Abre un archivo específico:

```
nano {{ruta/al/archivo}}
```

- Abre un archivo específico, posicionando el cursor en la línea y columna específica:

```
nano +{{línea}},{{columna}} {{ruta/al/archivo}}
```

- Abre un archivo específico y activa el ajuste de línea:

```
nano --softwrap {{ruta/al/archivo}}
```

- Abre un archivo específico y sangra nuevas líneas a la sangría de las líneas anteriores:

```
nano --autoindent {{ruta/al/archivo}}
```

- Abre nano y crea un archivo de resguardo (archivo~) cuando se guardan las ediciones:

```
nano --backup {{ruta/al/archivo}}
```

# nc

Redirige datos de entrada o salida a un flujo de red a través de esta versátil herramienta.

Más información: <https://manned.org/nc>.

- Inicia un escuchador en un puerto TCP y le envía un archivo:

```
nc -l -p {{puerto}} < {{nombre_de_archivo}}
```

- Conecta a un escuchador en un puerto y recibe un archivo de él:

```
nc {{host}} {{puerto}} > {{nombre_de_archivo_por_recibir}}
```

- Escanea los puertos TCP abiertos en un host:

```
nc -v -z -w {{tiempo_de_espera_en_segundos}} {{host}}  
{{puerto_inicial}}-{{puerto_final}}
```

- Inicia un escuchador en un puerto TCP y provee de acceso a tu intérprete de comandos local a la parte conectada (esto es peligroso y podría ser explotado):

```
nc -l -p {{puerto}} -e {{ejecutable_del_intérprete}}
```

- Conecta a un escuchador y provee de acceso a tu intérprete de comandos local a una parte remota (esto es peligroso y podría ser explotado):

```
nc {{host}} {{puerto}} -e {{ejecutable_del_intérprete}}
```

- Actúa como un proxy y envía información de un puerto TCP local a un host remoto:

```
nc -l -p {{puerto_local}} | nc {{host}} {{puerto_remoto}}
```

- Envía una petición HTTP GET:

```
echo -e "GET / HTTP/1.1\nHost: {{host}}\n\n" | nc {{host}} 80
```

# neo4j-admin

Gestiona y administra un Neo4j DBMS (Sistema de Gestión de Bases de Datos).

Más información: <https://neo4j.com/docs/operations-manual/current/tools/neo4j-admin/>.

- Inicia el DBMS:

```
neo4j-admin server start
```

- Detén el DBMS:

```
neo4j-admin server stop
```

- Establece la contraseña inicial del usuario predeterminada neo4j (requisito para el primer arranque del DBMS):

```
neo4j-admin dbms set-initial-password  
{{nombre_base_de_datos}}
```

- Crea un archivo con una base de datos sin conexión llamado nombre\_base\_de\_datos.dump:

```
neo4j-admin database dump --to-path={{ruta/al/directorio}}  
{{nombre_de_base_de_datos}}
```

- Carga una base de datos desde un archivo llamado nombre\_base\_de\_datos.dump:

```
neo4j-admin database load --from-path={{ruta/al/directorio}}  
{{nombre_de_base_de_datos}} --overwrite-destination=true
```

- Carga una base de datos desde un archivo especificado a través de stdin:

```
neo4j-admin database load --from-stdin  
{{nombre_de_base_de_datos}} --overwrite-destination=true <  
{{ruta/a/nombre_archivo.dump}}
```

- Muestra ayuda:

```
neo4j-admin --help
```



# nest

Herramienta de línea de comandos para inicializar, desarrollar y mantener aplicaciones Nest.

Más información: <https://docs.nestjs.com/cli/overview>.

- Muestra la información sobre la versión de nest instalada:

```
nest info
```

- Crea un nuevo proyecto NestJS en un directorio con el mismo nombre:

```
nest new {{nombre_del_proyecto}}
```

- Construye un proyecto NestJS específico:

```
nest build {{nombre_del_proyecto}}
```

- Ejecuta un proyecto NestJS específico:

```
nest start {{nombre_del_proyecto}}
```

- Importa una librería al proyecto NestJS actual:

```
nest add {{nombre_de_la_librería}}
```

# netlify

Despliegue de sitios y configuración del despliegue continuo en la plataforma de Netlify.

Más información: <https://cli.netlify.com>.

- Accede a la cuenta Netlify:

```
netlify login
```

- Despliega el contenido de un directorio en Netlify:

```
netlify deploy
```

- Configura el despliegue continuo para un sitio nuevo o existente:

```
netlify init
```

- Inicia un servidor local de desarrollo:

```
netlify dev
```

# next

Framework React que utiliza el renderizado del lado del servidor para crear aplicaciones web optimizadas.

Más información: <https://nextjs.org/docs>.

- Inicia la aplicación actual en modo de desarrollo:

```
next dev
```

- Inicia la aplicación actual y escucha en un puerto específico:

```
next dev --port {{puerto}}
```

- Construye la aplicación actual optimizada para producción:

```
next build
```

- Inicia la aplicación compilada en modo de producción:

```
next start
```

- Inicia la aplicación compilada y escucha en un puerto específico:

```
next start --port {{puerto}}
```

- Exporta la aplicación actual páginas HTML estáticas:

```
next export
```

- Muestra el estado de telemetría de Next.js:

```
next telemetry
```

- Muestra la ayuda de un subcomando:

```
next {{build|dev|export|start|telemetry}} --help
```

# nginx

Servidor web Nginx.

Más información: <https://nginx.org/en/>.

- Inicia el servidor con el archivo de configuración por defecto:

```
nginx
```

- Inicia el servidor con un archivo de configuración personalizado:

```
nginx -c {{archivo_de_configuración}}
```

- Inicia el servidor con un prefijo para todas las rutas relativas en el archivo de configuración:

```
nginx -c {{archivo_de_configuración}} -p {{prefijo/para/
rutas/relativas}}
```

- Prueba la configuración sin afectar al servidor en ejecución:

```
nginx -t
```

- Recarga la configuración enviando una señal sin tiempo de inactividad:

```
nginx -s reload
```

# nix classic

Una interfaz clásica y estable para un potente gestor de paquetes que hace la gestión de paquetes fiable, reproducible y declarativa.

Algunos comandos Nix como **nix-build**, **nix-shell**, **nix-env**, y **nix-store** tienen sus propias páginas. Ver también: **tldr nix**.

Más información: <https://nixos.org>.

- Busca un paquete en nixpkgs a través de su nombre:

```
nix-env -qaP {{termino_busqueda_regex}}
```

- Inicia un shell con los paquetes especificados disponibles:

```
nix-shell -p {{pkg1 pkg2 pkg3...}}
```

- Instala algunos paquetes de forma permanente:

```
nix-env -iA {{nixpkgs.pkg1 nixpkgs.pkg2...}}
```

- Muestra todas las dependencias de una ruta de almacenamiento (paquete), en formato de árbol:

```
nix-store --query --tree {/nix/store/...}}
```

- Actualiza los canales (repositorios):

```
nix-channel --update
```

- Elimina rutas no utilizadas del almacén Nix:

```
nix-collect-garbage
```

# nm-classic

Este comando es un alias de **nm**.

- Muestra la documentación del comando original:

```
tldr nm
```

# nmap

Herramienta de exploración de redes y escáner de seguridad/puertos.

Algunas funciones (p. ej. escaneo SYN) se activan solamente cuando **nmap** es ejecutado con los permisos del superusuario.

Más información: <https://nmap.org/book/man.html>.

- Escanea los top 1000 puertos de un host remoto con varios niveles de [v]erbosidad:

```
nmap -v{{1|2|3}} {{ip_o_nombre_de_host}}
```

- Ejecuta un barrido de ping en toda una subred o en hosts individuales de forma muy agresiva:

```
nmap -T5 -sn {{192.168.0.0/24|  
ip_o_nombre_de_host1,ip_o_nombre_de_host2,...}}
```

- Activa la detección de sistemas operativos, la detección de versión, el escaneo con secuencias de comandos y traceroute de hosts desde un archivo:

```
sudo nmap -A -iL {{ruta/al/archivo.txt}}
```

- Escanea una lista específica de puertos (usa -p- para todos los puertos de 1 a 65535):

```
nmap -p {{puerto1,puerto2,...}} {{ip_o_host1,ip_o_host2,...}}
```

- Detecta el servicio y versión de los top 1000 puertos usando las secuencias de comandos NSE por defecto y escribe los resultados (-oA) en archivos de salida:

```
nmap -sC -sV -oA {{top-1000-ports}}  
{{ip_o_host1,ip_o_host2,...}}
```

- Escanea objetivo(s) cuidadosamente usando las secuencias de comandos NSE default and safe:

```
nmap --script "default and safe"  
{{ip_o_host1,ip_o_host2,...}}
```

- Escanea servidores web ejecutándose en los puertos estándares 80 y 443 usando todas las secuencias de comando http-\* NSE disponibles:

```
nmap --script "http-*" {{ip_o_host1,ip_o_host2,...}} -p 80,443
```

- Intentar evadir la detección IDS/IPS utilizando un escaneo extremadamente lento (-T0) y usando direcciones de origen de señuelo (-D), paquetes [f]ragmentados, datos aleatorios y otros métodos:

```
sudo nmap -T0 -D {{ip_de_señuelo1,ip_de_señuelo2,...}} --source-port {{53}} -f --data-length {{16}} -Pn {{ip_o_host}}
```



# nms

Herramienta de línea de comandos que recrea el famoso efecto de desenscriptado de datos de la película Sneakers (1992).

Más información: <https://github.com/bartobri/no-more-secrets>.

- Desenscripta el texto tras presionar una tecla:

```
echo "{{Hola, Mundo!}}" | nms
```

- Desenscripta la salida inmediatamente, sin esperar a que una tecla sea pulsada:

```
{{ls -la}} | nms -a
```

- Desenscripta el contenido de un archivo, especificando el color de la salida:

```
cat {{ruta/al/archivo}} | nms -a -f {{blue|white|yellow|black|magenta|green|red}}
```

- Limpia la pantalla antes de desenscriptar:

```
{{comando}} | nms -a -c
```

# nnn

Gestor de archivos de terminal interactivo y analizador de uso de disco.

Más información: <https://github.com/jarun/nnn>.

- Abre el directorio actual (o especifica uno como primer argumento):

```
nnn
```

- Inicia en modo detallado:

```
nnn -d
```

- Muestra archivos ocultos:

```
nnn -H
```

- Abre un marcador existente (definido en la variable de entorno `NNN_BMS`):

```
nnn -b {{nombre_del_marcador}}
```

- Ordena los archivos por [a]parente uso de disco / uso de [d]isco / [e]xtensión / inve[r]so / tamaño / [t]iempo / [v]ersión:

```
nnn -T {{a|d|e|r|s|t|v}}
```

# npm

Gestor de paquetes JavaScript y Node.js.

Gestiona proyectos Node.js y sus dependencias de módulos.

Más información: <https://www.npmjs.com>.

- Crea un archivo `package.json` con los valores por defecto (omite `--yes` para hacerlo de forma interactiva):

```
npm init {{-y|--yes}}
```

- Descarga todos los paquetes listados como dependencias en `package.json`:

```
npm install
```

- Descarga una versión específica de un paquete y lo añade a la lista de dependencias en `package.json`:

```
npm install {{nombre_paquete}}@{{versión}}
```

- Descarga la última versión de un paquete y lo añade a la lista de dependencias de desarrollo en `package.json`:

```
npm install {{nombre_paquete}} {{-D|--save-dev}}
```

- Descarga la última versión de un paquete y lo instala globalmente:

```
npm install {{-g|--global}} {{nombre_paquete}}
```

- Desinstala un paquete y lo elimina de la lista de dependencias en `package.json`:

```
npm uninstall {{nombre_paquete}}
```

- Lista de dependencias instaladas localmente:

```
npm list
```

- Lista los paquetes instalados globalmente:

```
npm list {{-g|--global}} --depth {{0}}
```

# ntl

Este comando es un alias de **netlify**.

Más información: <https://cli.netlify.com>.

- Muestra la documentación del comando original:

```
tldr netlify
```

# nxc ftp

Prueba y explota servidores FTP.

Más información: <https://www.netexec.wiki/ftp-protocol>.

- Busca credenciales válidas probando cada combinación en las listas especificadas de nombres de [u]suario y contraseñas:

```
nxc ftp {{192.168.178.2}} -u {{ruta/a/nombres_de_usuario.txt}} -p {{ruta/a/contraseñas.txt}}
```

- Continúa buscando credenciales válidas incluso después de haberlas encontrado:

```
nxc ftp {{192.168.178.2}} -u {{ruta/a/nombres_de_usuario.txt}} -p {{ruta/a/contraseñas.txt}} --continue-on-success
```

- Realiza listados de directorios en cada servidor FTP en el que sean válidas las credenciales proporcionadas:

```
nxc ftp {{192.168.178.0/24}} -u {{nombre_de_usuario}} -p {{contraseña}} --ls
```

- Descarga el archivo especificado desde el servidor de destino:

```
nxc ftp {{192.168.178.2}} -u {{nombre_de_usuario}} -p {{contraseña}} --get {{ruta/al/archivo}}
```

- Carga el archivo especificado en el servidor de destino en la ubicación especificada:

```
nxc ftp {{192.168.178.2}} -u {{nombre_de_usuario}} -p {{contraseña}} --put {{ruta/al/archivo_local}} {{ruta/a/ubicación_remota}}
```

# nxc smb

Prueba y explota servidores SMB.

Más información: <https://www.netexec.wiki/smb-protocol>.

- Busca credenciales de dominio válidas probando cada combinación en las listas especificadas de nombres de [u]suario y contraseñas:

```
nxc smb {{192.168.178.2}} -u {{ruta/a/nombres_de_usuario.txt}} -p {{ruta/a/contraseñas.txt}}
```

- Busca credenciales válidas para cuentas locales en lugar de cuentas de dominio:

```
nxc smb {{192.168.178.2}} -u {{ruta/a/nombres_de_usuario.txt}} -p {{ruta/a/contraseñas.txt}} --local-auth
```

- Enumera los recursos compartidos SMB y los derechos de acceso de los usuarios especificados en los hosts de destino:

```
nxc smb {{192.168.178.0/24}} -u {{nombre_de_usuario}} -p {{contraseña}} --shares
```

- Enumera las interfaces de red en los hosts de destino, realizando la autenticación mediante pass-the-hash:

```
nxc smb {{192.168.178.30-45}} -u {{nombre_de_usuario}} -H {{NTLM_hash}} --interfaces
```

- Analiza los hosts de destino en busca de vulnerabilidades frecuentes:

```
nxc smb {{ruta/a/lista_objetivo.txt}} -u '' -p '' -M zerologon -M petitpotam
```

- Intenta ejecutar un comando en los hosts de destino:

```
nxc smb {{192.168.178.2}} -u {{nombre_de_usuario}} -p {{contraseña}} -x {{comando}}
```

# nxc ssh

Prueba y explota servidores SSH.

Vea también: **hydra**.

Más información: <https://www.netexec.wiki/ssh-protocol>.

- Pulveriza la contraseña especificada contra una lista de nombres de usuario en el objetivo especificado:

```
nxc ssh {{192.168.178.2}} -u {{ruta/a/nombres_de_usuario.txt}} -p {{contraseña}}
```

- Busca credenciales válidas probando todas las combinaciones de nombres de usuario y contraseñas de las listas especificadas:

```
nxc ssh {{192.168.178.2}} -u {{ruta/a/nombres_de_usuario.txt}} -p {{ruta/a/contraseñas.txt}}
```

- Utiliza la clave privada especificada para la autenticación, utilizando la contraseña suministrada como frase de contraseña de la clave:

```
nxc ssh {{192.186.178.2}} -u {{ruta/a/nombres_de_usuario.txt}} -p {{contraseña}} --key-file {{ruta/a/id_rsa}}
```

- Prueba una combinación de nombres de usuario y contraseñas en una serie de objetivos:

```
nxc ssh {{192.168.178.0/24}} -u {{nombre_de_usuario}} -p {{contraseña}}
```

- Comprueba los privilegios sudo en un inicio de sesión correcto:

```
nxc ssh {{192.168.178.2}} -u {{nombre_de_usuario}} -p {{ruta/a/contraseñas.txt}} --sudo-check
```

# nxc

Herramienta de enumeración y explotación de servicios de red.

Algunos subcomandos como **smb** tienen su propia documentación de uso.

Más información: <https://www.netexec.wiki/>.

- [L]ista módulos disponibles para el protocolo especificado:

```
nxc {{smb|ssh|ldap|ftp|wmi|winrm|rdp|vnc|mssql}} -L
```

- Lista las opciones disponibles para el módulo especificado:

```
nxc {{smb|ssh|ldap|ftp|wmi|winrm|rdp|vnc|mssql}} -M  
{{nombre_del_módulo}} --options
```

- Especifica una opción para un módulo:

```
nxc {{smb|ssh|ldap|ftp|wmi|winrm|rdp|vnc|mssql}} -M  
{{nombre_del_módulo}} -o {{NOMBRE_OPCION}}={{valor_opción}}
```

- Vea las opciones disponibles para el protocolo especificado:

```
nxc {{smb|ssh|ldap|ftp|wmi|winrm|rdp|vnc|mssql}} --help
```



# octez-client

Interactúa con la blockchain de Tezos.

Más información: <https://tezos.gitlab.io/introduction/howtouse.html#client>.

- Configura el cliente con una conexión a un nodo RPC de Tezos como <https://rpc.ghostnet.teztnets.com>:

```
octez-client -E {{endpoint}} config update
```

- Crea una cuenta y le asigna un alias local:

```
octez-client gen keys {{alias}}
```

- Obtén el saldo de una cuenta por alias o dirección:

```
octez-client get balance for {{alias_o_dirección}}
```

- Transfiere tez a otra cuenta:

```
octez-client transfer {{5}} from {{alias|address}} to  
{{alias|address}}
```

- Crea (despliega) un contrato inteligente, le asignar un alias local y establece su almacenamiento inicial como un valor codificado por Michelson:

```
octez-client originate contract {{alias}} transferring {{0}}  
from {{alias|address}} running {{ruta/a/  
archivo_de_origen.tz}} --init "{{almacenamiento_inicial}}" --  
burn_cap {{1}}
```

- Llama a un contrato inteligente por su alias o dirección y pasa un parámetro codificado por Michelson:

```
octez-client transfer {{0}} from {{alias|address}} to  
{{contract}} --entrypoint "{{entrypoint}}" --arg  
"{{parámetro}}" --burn-cap {{1}}
```

- Muestra ayuda:

```
octez-client man
```

# pambrighten

Cambia la saturación y el valor de una imagen PAM.

Más información: <https://netpbm.sourceforge.net/doc/pambrighten.html>.

- Aumenta la saturación de cada píxel con un porcentaje específico:

```
pambrighten -saturation {{valor_porcentual}} {{ruta/a/imagen.pam}} > {{ruta/a/archivo_de_salida.pam}}
```

- Aumenta el valor (del espacio de color HSV) de cada píxel con un porcentaje específico:

```
pambrighten -value {{valor_porcentual}} {{ruta/a/imagen.pam}} > {{ruta/a/archivo_de_salida.pam}}
```

# pamedge

Realiza la detección de bordes en una imagen Netpbm.

Más información: <https://netpbm.sourceforge.net/doc/pamedge.html>.

- Detecta bordes en una imagen Netpbm:

```
pamedge {{ruta/a/entrada.pam}} > {{ruta/a/salida.pam}}
```

# pamslice

Extrae una línea de valores de una imagen PAM.

Más información: <https://netpbm.sourceforge.net/doc/pamslice.html>.

- Imprime los valores de los píxeles de la n-ésima fila en una tabla:

```
pamslice -row {{n}} {{ruta/a/imagen.pam}}
```

- Imprime los valores de los píxeles de la n-ésima columna de una tabla:

```
pamslice -column {{n}} {{ruta/a/imagen.pam}}
```

- Considera solo el m-ésimo plano de la imagen de entrada:

```
pamslice -row {{n}} -plane {{m}} {{ruta/a/imagen.pam}}
```

- Produce la salida en un formato adecuado para la entrada a un xmgr para la visualización:

```
pamslice -row {{n}} -xmgr {{ruta/a/imagen.pam}}
```

# pamtogif

Convierte una imagen Netpbm en una imagen GIF no animada.

Vea también: **giftopnm**, **gifsicle**.

Más información: <https://netpbm.sourceforge.net/doc/pamtogif.html>.

- Convierte una imagen Netpbm en una imagen GIF no animada:

```
pamtogif {{ruta/a/imagen.pam}} > {{ruta/a/imagen_de_salida.gif}}
```

- Marca el color especificado como transparente en el archivo GIF de salida:

```
pamtogif -transparent {{color}} {{ruta/a/imagen.pam}} > {{ruta/a/imagen_de_salida.gif}}
```

- Incluye el texto especificado como comentario en el archivo GIF de salida:

```
pamtogif -comment "{{¡Hola Mundo!}}" {{ruta/a/imagen.pam}} > {{ruta/a/imagen_de_salida.gif}}
```

# pamtouil

Convierte un archivo PNM o PAM en un archivo de iconos UIL de Motif.

Más información: <https://netpbm.sourceforge.net/doc/pamtouil.html>.

- Convierte un archivo PNM o PAM en un archivo de icono Motif UIL:

```
pamtouil {{ruta/a/entrada.[pnm|pam]}} > {{ruta/a/salida.uil}}
```

- Especifique una cadena de prefijo que se imprimirá en el archivo UIL de salida:

```
pamtouil -name {{nombre_uil}} {{ruta/a/entrada.[pnm|pam]}} > {{ruta/a/salida.uil}}
```

# pants

Herramienta de flujo de trabajo rápida, escalable, fácil de usar y de código abierto.

Más información: <https://www.pantsbuild.org/2.20/docs/using-pants/command-line-help>.

- Lista todos los objetivos:

```
pants list ::
```

- Ejecuta todas las pruebas:

```
pants test ::
```

- Arregla, formatea y limpia sólo los archivos no comprometidos:

```
pants --changed-since=HEAD fix fmt lint
```

- Comprueba sólo los archivos no comprometidos y sus dependientes:

```
pants --changed-since=HEAD --changed-dependents=transitive  
check
```

- Crea un paquete distribuible para el objetivo especificado:

```
pants package {{ruta/al/directorio:nombre-destino}}
```

- Autogenera objetivos de archivo BUILD para nuevos archivos fuente:

```
pants tailor ::
```

- Muestra la ayuda:

```
pants help
```

# par2

Verificación y reparación de archivos utilizando archivos de paridad compatibles con PAR 2.0 (archivos .par2).

Más información: <https://github.com/Parchive/par2cmdline/>.

- Crea un archivo de paridad con un nivel de porcentaje de redundancia establecido:

```
par2 create -r{{1..100}} -- {{ruta/al/archivo}}
```

- Crea un archivo de paridad con un número determinado de archivos de volumen (además del archivo de índice):

```
par2 create -n{{1..32768}} -- {{ruta/al/archivo}}
```

- Verifica un fichero con un archivo de paridad:

```
par2 verify -- {{ruta/al/archivo.par2}}
```

- Repara un fichero con un archivo de paridad:

```
par2 repair -- {{ruta/al/archivo.par2}}
```



# passwd

Cambia la contraseña de un usuario.

Más información: <https://manned.org/passwd>.

- Cambia la contraseña del usuario actual de forma interactiva:

```
passwd
```

- Cambia la contraseña de un usuario específico:

```
passwd {{nombre_usuario}}
```

- Obtiene el estado actual del usuario:

```
passwd {{-S|--status}}
```

- Hace que la contraseña de la cuenta esté en blanco (hará que la cuenta nombrada no tenga contraseña):

```
passwd {{-d|--delete}}
```

# pcapfix

Repara archivos PCAP y PcapNG dañados o corruptos.

Más información: <https://f00l.de/pcapfix/>.

- Repara un archivo PCAP/PcapNG (Nota: para los archivos PCAP, sólo se escanean los primeros 262144 bytes de cada paquete):

```
pcapfix {{ruta/al/archivo.pcapng}}
```

- Repara un archivo PCAP completo:

```
pcapfix --deep-scan {{ruta/al/archivo.pcap}}
```

- Repara un archivo PCAP/PcapNG y escribe el archivo reparado en la ubicación especificada:

```
pcapfix --outfile {{ruta/al/archivo_reparado.pcap}} {{ruta/al/archivo.pcap}}
```

- Repara un archivo PcapNG y lo trata como un archivo PcapNG, ignorando el reconocimiento automático:

```
pcapfix --pcapng {{ruta/al/archivo.pcapng}}
```

- Repara un archivo y muestra el proceso en detalle:

```
pcapfix --verbose {{ruta/al/archivo.pcap}}
```

# pcdindex

Este comando ha sido renombrado a **pcdovtoppm**.

Más información: <https://netpbm.sourceforge.net/doc/pcdindex.html>.

- Consulta la documentación del comando con su nombre actual:

```
tldr pcdovtoppm
```

# pcdovtoppm

Crea una imagen índice para un CD de fotos basándose en su archivo de resumen.

Más información: <https://netpbm.sourceforge.net/doc/pcdovtoppm.html>.

- Crea una imagen índice PPM a partir de un archivo de vista general PCD:

```
pcdovtoppm {{ruta/al/archivo.pcd}} > {{ruta/al/archivo.ppm}}
```

- Especifica la anchura [m]áxima de la imagen de salida y el tamaño máximo de cada una de las imágenes contenidas en la salida:

```
pcdovtoppm -m {{anchura}} -s {{tamaño}} {{ruta/al/archivo.pcd}} > {{ruta/al/archivo.ppm}}
```

- Especifica el número máximo de imágenes y el número máximo de [c]olores:

```
pcdovtoppm -a {{n_imágenes}} -c {{n_colores}} {{ruta/al/archivo.pcd}} > {{ruta/al/archivo_salida.ppm}}
```

- Utiliza la [f]uente especificada para las anotaciones y pinta el fondo blanco:

```
pcdovtoppm -f {{fuente}} -w {{ruta/al/archivo.pcd}} > {{ruta/al/archivo.ppm}}
```

# pgmedge

Este comando ha sido sustituido por **pamedge**.

Más información: <https://netpbm.sourceforge.net/doc/pgmedge.html>.

- Consulta la documentación del comando actual:

```
tldr pamedge
```

# pgmnorm

Este comando es reemplazado por **pnmnorm**.

Más información: <https://netpbm.sourceforge.net/doc/pgmnorm.html>.

- Muestra la documentación del comando actual:

```
tldr pnmnorm
```

# pgmslice

Este comando ha sido sustituido por **pamslice**.

Más información: <https://netpbm.sourceforge.net/doc/pgmslice.html>.

- Vea documentación para el comando actual:

```
tldr pamslice
```

# pgmtopbm

Este comando ha sido sustituido por **pamditherbw**.

Más información: <https://netpbm.sourceforge.net/doc/pgmtopbm.html>.

- Vea documentación del comando actual:

```
tldr pamditherbw
```



# pi1toppm

Convierte una imagen Atari Degas PI1 en una imagen PPM.

Vea también: **ppmtopi1**.

Más información: <https://netpbm.sourceforge.net/doc/pi1toppm.html>.

- Convierte una imagen Atari Degas PI1 en una imagen PPM:

```
pi1toppm {{ruta/a/imagen_atari.pi1}} > {{ruta/a/imagen.ppm}}
```

# picocom

Programa minimalista para emular consolas serie.

Más información: <https://manned.org/picocom>.

- Conecta a una consola serie con una velocidad en baudios específica:

```
picocom {{/dev/ttyXYZ}} --baud {{tasa_de_baudios}}
```

- Asigna caracteres especiales (p. ej. LF a CRLF):

```
picocom {{/dev/ttyXYZ}} --imap {{lfcrlf}}
```

# ping

Envía paquetes ICMP ECHO\_REQUEST (pings) a hosts de la red.

Más información: <https://manned.org/ping>.

- Envía pings a un host:

```
ping {{host}}
```

- Envía un número determinado de pings a un host:

```
ping -c {{numero}} {{host}}
```

- Envía pings a un host especificando el intervalo de tiempo entre peticiones (por defecto 1 segundo):

```
ping -i {{segundos}} {{host}}
```

- Envía pings a un host sin intentar resolver nombres simbólicos de direcciones:

```
ping -n {{host}}
```

- Envía pings a un host y emite un sonido cuando un paquete es recibido (si la terminal lo soporta):

```
ping -a {{host}}
```

- Muestra también un mensaje si no se recibió respuesta:

```
ping -0 {{host}}
```

# ping6

Envía paquetes ICMP ECHO\_REQUEST (pings) a hosts de la red usando direcciones IPv6.

Más información: <https://manned.org/ping6>.

- Envía pings a un host:

```
ping6 {{host}}
```

- Envía un número específico de pings a un host:

```
ping6 -c {{numero}} {{host}}
```

- Envía pings a un host, especificando el intervalo de tiempo entre peticiones (por defecto es 1 segundo):

```
ping6 -i {{segundos}} {{host}}
```

- Envía pings a un host sin intentar resolver nombres simbólicos de direcciones:

```
ping6 -n {{host}}
```

- Envía pings a un host y emite un sonido cuando un paquete es recibido (si la terminal lo soporta):

```
ping6 -a {{host}}
```

# pio init

Este comando es un alias de **pio project**.

- Muestra la documentación del comando original:

```
tldr pio project
```

# piodebuggdb

Este comando es un alias de **pio debug**.

- Muestra la documentación del comando original:

```
tldr pio debug
```

# piper

Un sistema neural rápido y local de conversión de texto a voz.

Descarga y prueba modelos de habla desde <https://rhasspy.github.io/piper-samples>.

Más información: <https://github.com/rhasspy/piper>.

- Genera un archivo WAV utilizando un modelo de texto a voz (suponiendo un archivo de configuración en model\_path + .json):

```
echo {{Cosa a decir}} | piper -m {{ruta/a/modelo.onnx}} -f {{archivo_de_salida.wav}}
```

- Genera un archivo WAV utilizando un modelo y especificando su archivo de [c]onfiguración JSON:

```
echo {{'Lo que hay que decir'}} | piper -m {{ruta/a/modelo.onnx}} -c {{ruta/a/modelo.onnx.json}} -f {{archivo_de_salida.wav}}
```

- Selecciona un locutor concreto en una voz con varios locutores especificando el número de identificación del locutor:

```
echo {{'Warum?'}} | piper -m {{de_DE-thorsten_emotional-medium.onnx}} --speaker {{1}} -f {{enojado.wav}}
```

- Transmite la salida al reproductor multimedia mpv:

```
echo {{'Hello world'}} | piper -m {{en_GB-northern_english_male-medium.onnx}} --output-raw -f - | mpv -
```

- Habla el doble de rápido, con grandes espacios entre frases:

```
echo {{'Hablando el doble de rápido. Con más drama!'}} | piper -m {{foo.onnx}} --length_scale {{0.5}} --sentence_silence {{2}} -f {{drama.wav}}
```

# plantuml

Crea diagramas UML a partir de un lenguaje de texto plano y los renderiza en diferentes formatos.

Más información: <https://plantuml.com/en/command-line>.

- Renderiza los diagramas al formato por defecto (PNG):

```
plantuml {{diagrama1.puml}} {{diagrama2.puml}}
```

- Renderiza un diagrama en un formato determinado (p.ej. png, pdf, svg, txt):

```
plantuml -t {{formato}} {{diagrama.puml}}
```

- Renderiza todos los diagramas de un directorio:

```
plantuml {{ruta/a/diagramas}}
```

- Renderiza un diagrama al directorio de salida:

```
plantuml -o {{ruta/a/salida}} {{diagrama.puml}}
```

- Renderiza un diagrama sin almacenar el código fuente del diagrama (Nota: Se almacena por defecto cuando no se especifica la opción -nometadata):

```
plantuml -nometadata {{diagrama.png}} > {{diagrama.puml}}
```

- Recupera la fuente de los metadatos de un diagrama plantuml:

```
plantuml -metadata {{diagrama.png}} > {{diagrama.puml}}
```

- Renderiza un diagrama con el archivo de configuración:

```
plantuml -config {{config.cfg}} {{diagrama.puml}}
```

- Muestra la ayuda:

```
plantuml -help
```



# platformio

Este comando es un alias de **pio**.

Más información: <https://docs.platformio.org/en/latest/core/userguide/>.

- Muestra la documentación del comando original:

```
tldr pio
```

# pngtopnm

Este comando ha sido sustituido por **pngtopam**.

Más información: <https://netpbm.sourceforge.net/doc/pngtopnm.html>.

- Ve documentación del comando actual:

```
tldr pngtopam
```

# pnmcut

Este comando ha sido sustituido por **pamcut**.

Más información: <https://netpbm.sourceforge.net/doc/pnmcut.html>.

- Ve documentación del comando actual:

```
tldr pamcut
```

# pnmenlarge

Este comando ha sido sustituido por **pamenlarge**.

Más información: <https://netpbm.sourceforge.net/doc/pnmenlarge.html>.

- Vea documentación del comando actual:

```
tldr pamenlarge
```

# pnmfile

Este comando ha sido sustituido por **pamfile**.

Más información: <https://netpbm.sourceforge.net/doc/pnmfile.html>.

- Vea la documentación del comando actual:

```
tldr pamfile
```

# pnmflip

Este comando ha sido sustituido por **pamflip**.

Más información: <https://netpbm.sourceforge.net/doc/pnmflip.html>.

- Vea documentación del comando actual:

```
tldr pamflip
```

# pnminterp

Este comando ha sido sustituido por **pamstretch**.

Más información: <https://netpbm.sourceforge.net/doc/pnminterp.html>.

- Ve documentación del comando actual:

```
tldr pamstretch
```

# pnmpsnr

Calcula la diferencia entre dos imágenes.

Más información: <https://netpbm.sourceforge.net/doc/pnmpsnr.html>.

- Calcula la diferencia, es decir, la relación señal-ruido (PSNR) entre dos imágenes:

```
pnmpsnr {{ruta/al/archivo1.pnm}} {{ruta/al/archivo2.pnm}}
```

- Compara los componentes de color en lugar de los componentes de luminancia y crominancia de las imágenes:

```
pnmpsnr {{ruta/al/archivo1.pnm}} {{ruta/al/archivo2.pnm}} -  
rgb
```

- Ejecuta en modo de comparación, es decir, solo la salida nomatch o match dependiendo de si el cálculo PSNR supera n o no:

```
pnmpsnr {{ruta/al/archivo1.pnm}} {{ruta/al/archivo2.pnm}} -  
target {{n}}
```

- Ejecuta en modo comparación y compara los componentes individuales de la imagen, es decir, Y, Cb y Cr, con los umbrales correspondientes:

```
pnmpsnr {{ruta/al/archivo1.pnm}} {{ruta/al/archivo2.pnm}} -  
target1 {{umbral_Y}} -target2 {{umbral_Cb}} -target3  
{{umbral_Cr}}
```

- Ejecuta en modo comparación y compara los componentes individuales de la imagen, es decir, rojo, verde y azul con los umbrales correspondientes:

```
pnmpsnr {{ruta/al/archivo1.pnm}} {{ruta/al/archivo2.pnm}} -  
rgb -target1 {{umbral_rojo}} -target2 {{umbral_verde}} -  
target3 {{umbral_azul}}
```

- Produce salida legible para máquinas:

```
pnmpsnr {{ruta/al/archivo1.pnm}} {{ruta/al/archivo2.pnm}} -  
machine
```



# pnmtjpeg

Convierte un archivo de imagen PNM al formato de imagen JPEG/JFIF/EXIF.

Más información: <https://netpbm.sourceforge.net/doc/pnmtjpeg.html>.

- Lee una imagen PNM como entrada y produce una imagen JPEG/JFIF/EXIF como salida:

```
pnmtjpeg {{ruta/al/archivo.pnm}} > {{ruta/al/archivo.jpg}}
```

- Muestra la versión:

```
pnmtjpeg -version
```

# ppmbrighten

Este comando ha sido sustituido por **pambrighten**.

Más información: <https://netpbm.sourceforge.net/doc/ppmbrighten.html>.

- Vea documentación del comando actual:

```
tldr pambrighten
```

# ppmnorm

Este comando es reemplazado por **pnmnorm**.

Más información: <https://netpbm.sourceforge.net/doc/ppmnorm.html>.

- Muestra la documentación del comando actual:

```
tldr pnmnorm
```

# ppmtogif

Este comando ha sido sustituido por **pamtogif**.

Más información: <https://netpbm.sourceforge.net/doc/ppmtogif.html>.

- Vea documentación para el comando actual:

```
tldr pamtogif
```

# ppmtojpeg

Este comando ha sido sustituido por **pnmtjpeg**.

Más información: <https://netpbm.sourceforge.net/doc/ppmtojpeg.html>.

- Ve documentación del comando actual:

```
tldr pnmtjpeg
```

# ppmtomap

Este comando ha sido sustituido por **pnmcolormap**.

Más información: <https://netpbm.sourceforge.net/doc/ppmtomap.html>.

- Vea documentación del comando actual:

```
tldr pnmcolormap
```

# ppmtopi1

Convierte una imagen PPM en una imagen Atari Degas PI1.

Vea también: **pi1toppm**.

Más información: <https://netpbm.sourceforge.net/doc/ppmtopi1.html>.

- Convierte una imagen PPM en una imagen Atari Degas PI1:

```
ppmtopi1 {{ruta/a/imagen.ppm}} > {{ruta/a/imagen_salida.pi1}}
```

# ppmtouil

Este comando ha sido sustituido por **pamtouil**.

Más información: <https://netpbm.sourceforge.net/doc/ppmtouil.html>.

- Consulte la documentación del comando actual:

```
tldr pamtouil
```



# ps

Información sobre procesos en ejecución.

Más información: <https://manned.org/ps>.

- Lista todos los procesos en ejecución:

```
ps aux
```

- Lista todos los procesos en ejecución incluyendo el comando completo:

```
ps auxww
```

- Busca un proceso que coincida con la cadena de texto:

```
ps aux | grep {{cadena}}
```

- Lista todos los procesos del usuario actual en formato supercompleto:

```
ps --user $(id -u) -F
```

- Lista todos los procesos del usuario actual como un árbol:

```
ps --user $(id -u) f
```

- Obtén el PID del proceso padre:

```
ps -o ppid= -p {{pid}}
```

- Ordena los procesos por consumo de memoria:

```
ps --sort size
```

# ptpython3

Este comando es un alias de **ptpython**.

- Muestra la documentación del comando original:

```
tldr ptpython
```

# pulumi login

Inicia sesión en Pulumi Cloud.

Más información: [https://www.pulumi.com/docs/cli/commands/pulumi\\_login/](https://www.pulumi.com/docs/cli/commands/pulumi_login/).

- Inicia sesión en el servidor administrado Pulumi Cloud, de manera predeterminada en `app.pulumi.cloud`:

```
pulumi login
```

- Inicia sesión en un backend Pulumi Cloud autoalojado en una URL especificada:

```
pulumi login {{url}}
```

- Utiliza Pulumi localmente, independientemente de Pulumi Cloud:

```
pulumi login {{-l|--local}}
```

# pwd

Muestra el nombre del directorio actual.

Más información: <https://www.gnu.org/software/coreutils/pwd>.

- Muestra el directorio actual:

```
pwd
```

- Muestra el directorio actual y resuelve todos los enlaces simbólicos (es decir, muestra la ruta "física"):

```
pwd -P
```

# pydocstyle

Comprueba estáticamente que los scripts de Python cumplen con las convenciones de documentación de Python.

Más información: <https://www.pydocstyle.org/en/latest/>.

- Analiza un script de Python o todos los scripts de Python en un directorio específico:

```
pydocstyle {{archivo.py|ruta/al/directorio}}
```

- Muestra una explicación de cada error:

```
pydocstyle {{-e|--explain}} {{archivo.py|ruta/al/directorio}}
```

- Muestra información de depuración:

```
pydocstyle {{-d|--debug}} {{archivo.py|ruta/al/directorio}}
```

- Muestra el número total de errores:

```
pydocstyle --count {{archivo.py|ruta/a/directorio}}
```

- Utiliza un archivo de configuración específico:

```
pydocstyle --config {{ruta/a/archivo_config}} {{archivo.py|  
ruta/al/directorio}}
```

- Ignora uno o más errores:

```
pydocstyle --ignore {{D101,D2,D107,...}} {{archivo.py|ruta/  
al/directorio}}
```

- Busca errores de una convención específica:

```
pydocstyle --convention {{pep257|numpy|google}} {{archivo.py|  
ruta/a/directorio}}
```

# python3

Este comando es un alias de **python**.

- Muestra la documentación del comando original:

```
tldr python
```

# qalc

Calculadora de línea de comandos potente y fácil de usar.

Vea también: **bc**.

Más información: <https://qalculate.github.io/manual/qalc.html>.

- Lanzamiento en modo [i]nteractivo:

```
qalc {{--interactive}}
```

- Ejecuta en modo [t]erse (solo imprime los resultados):

```
qalc --terse
```

- Actualiza los tipos de cambio:

```
qalc --exrates
```

- Realiza cálculos de forma no interactiva:

```
qalc {{66+99|2^4|6 pies a cm|1 bitcoin a USD|20 kmph a mph|...}}
```

- Lista todas las funciones/prefijos/unidades/variables soportadas:

```
qalc --{{list-functions|list-prefixes|list-units|list-variables}}
```

- Ejecuta comandos desde un archivo:

```
qalc --file {{ruta/a/archivo}}
```

# qc

Gestiona y ejecuta fragmentos de comandos almacenados en notas QOwnNotes.

Ve también: **qownnotes**.

Más información: <https://www.qownnotes.org/getting-started/command-line-snippet-manager.html>.

- Configura el gestor de fragmentos, por ejemplo para establecer el token de seguridad de QOwnNotes:

```
qc configure
```

- Busca e imprime fragmentos de comandos almacenados en tu nota `Commands.md` y en todas tus notas etiquetadas con `commands`:

```
qc search
```

- Ejecuta un fragmento y muestra el comando antes de ejecutarlo:

```
qc exec --command
```

- Ejecuta el último fragmento y muestra el comando antes de ejecutarlo:

```
qc exec --command --last
```

- Cambia entre las carpetas de notas en QOwnNotes:

```
qc switch
```



# qownnotes

Aplicación de toma de notas en formato Markdown.

Se integra opcionalmente con las aplicaciones de toma de notas de Nextcloud y ownCloud.

Vea también: **qc**, para gestionar fragmentos de comandos.

Más información: <https://www.qownnotes.org/getting-started/cli-parameters.html>.

- Ejecuta en modo portable:

```
QOwnNotes --portable
```

- Muestra la configuración y cualquier otra información sobre la aplicación y su entorno en formato GitHub Markdown:

```
QOwnNotes --dump-settings
```

- Especifica un contexto diferente para la configuración y los archivos internos:

```
QOwnNotes --session {{prueba}}
```

- Activa una acción de menú después de iniciar la aplicación:

```
QOwnNotes --action {{actionShow_Todo_List}}
```

# quarkus

Crea proyectos Quarkus, gestiona extensiones y realiza tareas esenciales de compilación y desarrollo.

Más información: <https://quarkus.io/guides/cli-tooling>.

- Crea un nuevo proyecto de aplicación en un directorio nuevo:

```
quarkus create app {{nombre_del_proyecto}}
```

- Ejecuta el proyecto actual en el modo de codificación en vivo:

```
quarkus dev
```

- Ejecuta la aplicación:

```
quarkus run
```

- Ejecuta el proyecto actual en modo de prueba continua:

```
quarkus test
```

- Añade una o más extensiones al proyecto actual:

```
quarkus extension add {{nombre_extensión1 nombre_extensión2  
...}}
```

- Construye un contenedor de imagen utilizando Docker:

```
quarkus image build docker
```

- Despliega la aplicación en Kubernetes:

```
quarkus deploy kubernetes
```

- Actualiza el proyecto:

```
quarkus update
```

# r2

Este comando es un alias de **radare2**.

- Muestra la documentación del comando original:

```
tldr radare2
```

# rcat

Este comando es un alias de **rc**.

- Muestra la documentación del comando original:

```
tldr rc
```

# read

Shell builtin para recuperar datos de **stdin**.

Más información: <https://manned.org/read.1p>.

- Almacena los datos que escribes desde el teclado:

```
read {{variable}}
```

- Almacena cada una de las siguientes líneas que introduzcas como valores de un arreglo:

```
read -a {{arreglo}}
```

- Especifica el número máximo de caracteres a leer:

```
read -n {{cuenta_caracteres}} {{variable}}
```

- Asigna varios valores a varias variables:

```
read {{_ variable1 _ variable2}} <<< {"El apellido es Bond"}}
```

- No dejes que la barra invertida () actúe como carácter de escape:

```
read -r {{variable}}
```

- Muestra un indicador antes de la entrada:

```
read -p "{{Introduce aquí tu entrada: }}" {{variable}}
```

- No hacer eco de los caracteres introducidos (modo silencioso):

```
read -s {{variable}}
```

- Lee **stdin** y realiza una acción en cada línea:

```
while read line; do {{echo|ls|rm|...}} "$line"; done < {{dev/  
stdin|ruta/al/archivo|...}}
```

# ripgrep

Este comando es un alias de **rg**.

- Muestra la documentación del comando original:

```
tldr rg
```

# rkdeveloptool

Flashea, vacía y gestiona el firmware de arranque en dispositivos informáticos basados en Rockchip.

Necesitará encender el dispositivo en modo Maskrom/Bootrom antes de conectarlo a través del USB.

Algunos subcomandos pueden requerir ser ejecutados como el superusuario.

Más información: <https://github.com/rockchip-linux/rkdeveloptool>.

- [l]ista todos los [d]ispositivos flash conectados basados en Rockchip:

```
rkdeveloptool ld
```

- Inicializa el dispositivo forzándolo a [d]escargar e instalar el gestor de arranque desde un archivo:

```
rkdeveloptool db {{ruta/al/bootloader.bin}}
```

- Actualiza un gestor de arranque:

```
rkdeveloptool ul {{ruta/al/bootloader.bin}}
```

- Escribe una imagen en una partición flash con formato GPT, especificando el sector de almacenamiento inicial (normalmente 0x0, alias 0):

```
rkdeveloptool wl {{sector_inicial}} {{ruta/a/imagen.img}}
```

- Escribe en la partición flash por su nombre amigable:

```
rkdeveloptool wlx {{nombre_partición}} {{ruta/a/imagen.img}}
```

- [r]einicia/repón el [d]ispositivo, sal del modo Maskrom/Bootrom para arrancar en la partición flash seleccionada:

```
rkdeveloptool rd
```

# rm

Elimina archivos o directorios.

Más información: <https://www.gnu.org/software/coreutils/rm>.

- Elimina archivos de ubicaciones arbitrarias:

```
rm {{ruta/al/archivo1 ruta/al/archivo2 ...}}
```

- Elimina varios archivos de forma interactiva, solicitando confirmación antes de eliminar cada archivo:

```
rm -i {{archivo(s)}}
```

- Elimina archivos en modo detallado, imprimiendo un mensaje por cada archivo eliminado:

```
rm -v {{ruta/hacia/directorio/*}}
```

- Elimina, de forma recursiva, un directorio y todos sus subdirectorios:

```
rm -r {{ruta/al/directorio}}
```



# rubocop

Analiza archivos de Ruby.

Más información: [https://docs.rubocop.org/rubocop/usage/basic\\_usage.html](https://docs.rubocop.org/rubocop/usage/basic_usage.html).

- Verifica todos los archivos en el directorio actual (incluyendo subdirectorios):

```
rubocop
```

- Verifica uno o más archivos o directorios determinados:

```
rubocop {{ruta/al/archivo_o_directorio1 ruta/al/
archivo_o_directorio2 ...}}
```

- Guarda la salida en un archivo:

```
rubocop --out {{ruta/al/archivo}}
```

- Muestra la lista de cops (reglas de análisis):

```
rubocop --show-cops
```

- Excluye una regla:

```
rubocop --except {{cop1 cop2 ...}}
```

- Ejecuta solo determinadas reglas:

```
rubocop --only {{cop1 cop2 ...}}
```

- Autocorriga archivos (experimental):

```
rubocop --auto-correct
```

# rustscan

Escáner de puertos rápido, escrito en Rust integrado con **nmap**.

Más información: <https://github.com/RustScan/RustScan>.

- Escanea todos los puertos de una o más direcciones delimitadas por comas usando los valores predeterminados:

```
rustscan --addresses {{ip_o_nombrehost}}
```

- Escanea los [t]op 1000 puertos con detección de servicio y versión:

```
rustscan --top --addresses {{dirección_o_direcciones}}
```

- Escanea una lista específica de [p]uertos:

```
rustscan --ports {{puerto1,puerto2,...,puertoN}} --addresses {{dirección_o_direcciones}}
```

- Escanea un rango específico de puertos:

```
rustscan --range {{inicio-fin}} --addresses {{dirección_o_direcciones}}
```

- Añade argumentos de script a nmap:

```
rustscan --addresses {{dirección_o_direcciones}} -- -A -sC
```

- Escanea con un tamaño de lote ([b]atch) (por defecto: 4500) y [t]iempo de espera personalizado (por defecto: 1500ms):

```
rustscan --batch-size {{tamaño_lote}} --timeout {{timeout}} --addresses {{dirección_o_direcciones}}
```

- Escanea puertos en un orden específico:

```
rustscan --scan-order {{serial|random}} --addresses {{dirección_o_direcciones}}
```

- Escanea en modo "greppable" (solo imprime los puertos y no usa nmap):

```
rustscan --greppable --addresses {{dirección_o_direcciones}}
```

# shotcut

Un programa para la edición de vídeo.

Más información: <https://shotcut.org/notes/command-line-options/>.

- Inicia Shotcut:

```
shotcut
```

- Abre archivos de audio/vídeo:

```
shotcut {{ruta/al/archivo1 ruta/al/archivo2 ...}}
```

- Inicia con un controlador de audio específico:

```
shotcut --SDL_AUDIODRIVER "{{pulseaudio}}"
```

- Inicia en pantalla completa:

```
shotcut --fullscreen
```

- Inicia con procesamiento GPU:

```
shotcut --gpu
```

# showfigfonts

Muestra una lista de fuentes disponibles para figlet.

Vea también **figlet**.

Más información: <https://manned.org/showfigfonts>.

- Muestra las fuentes disponibles:

```
showfigfonts
```

- Muestra las fuentes disponibles usando un texto específico:

```
showfigfonts {{texto_de_entrada}}
```

# simplehttpserver

Un simple servidor HTTP/S que soporta subida de ficheros, autenticación básica y reglas YAML para respuestas personalizadas.

Una alternativa Go a **http.server** de Python.

Más información: <https://github.com/projectdiscovery/simplehttpserver>.

- Inicia el servidor HTTP que sirve el directorio actual con salida verbosa (escucha en todas las interfaces y puerto 8000 por defecto):

```
simplehttpserver -verbose
```

- Inicia el servidor HTTP con autenticación básica sirviendo una ruta específica a través del puerto 80 en todas las interfaces:

```
sudo simplehttpserver -basic-auth {{nombre_de_usuario}}:{{contraseña}} -path {{/var/www/html}} -listen 0.0.0.0:80
```

- Inicia el servidor HTTP, habilitando HTTPS utilizando un certificado autofirmado con SAN personalizado en todas las interfaces:

```
sudo simplehttpserver -https -domain {{*.selfsigned.com}} -listen 0.0.0.0:443
```

- Inicia el servidor HTTP con cabeceras de respuesta personalizadas y capacidad de carga:

```
simplehttpserver -upload -header '{{X-Powered-By: Go}}' -header '{{Server: SimpleHTTPServer}}'
```

- Inicia el servidor HTTP con reglas personalizables en YAML (consulte la documentación para DSL):

```
simplehttpserver -rules {{rules.yaml}}
```

# streamlit

Marco de aplicación para crear aplicaciones web interactivas y basadas en datos en Python.

Más información: <https://docs.streamlit.io/>.

- Comprueba la instalación de Streamlit:

```
streamlit hello
```

- Ejecuta una aplicación Streamlit:

```
streamlit run {{nombre_del_proyecto}}
```

- Muestra ayuda:

```
streamlit --help
```

- Muestra la versión:

```
streamlit --version
```

# tail

Muestra las últimas líneas de un archivo de texto determinado.

Vea también: **head**.

Más información: <https://www.gnu.org/software/coreutils/tail>.

- Imprime las últimas líneas de 'recuento' de un archivo:

```
tail --lines {{recuento}} {{ruta/al/archivo}}
```

- Imprime un archivo desde una línea específica:

```
tail --lines +{{recuento}} {{ruta/al/archivo}}
```

- Imprime un número específico de bytes desde el final de algún archivo:

```
tail --bytes {{recuento}} {{ruta/al/archivo}}
```

- Imprime las últimas líneas de un archivo en tiempo real hasta presionar Ctrl + C:

```
tail --follow {{ruta/al/archivo}}
```

- Mantiene leyendo las últimas líneas de un archivo hasta presionar Ctrl + C, aunque el archivo sea inaccesible:

```
tail --retry --follow {{ruta/al/archivo}}
```

- Imprime las últimas líneas de 'recuento' en 'archivo' y se actualiza cada 'n' segundos:

```
tail --lines {{recuento}} --sleep-interval {{segundos}} --follow {{ruta/al/archivo}}
```

# tar

Utilidad de archivado.

A menudo se combina con un método de compresión, como gzip o bzip2.

Más información: <https://www.gnu.org/software/tar>.

- [c]rea un archivo y lo escribe en un [f]ichero:

```
tar cf {{ruta/al/objetivo.tar}} {{ruta/al/archivo1 ruta/al/archivo2 ...}}
```

- [c]rea un archivo con formato g[z]ip y lo escribe en un [f]ichero:

```
tar czf {{ruta/al/objetivo.tar.gz}} {{ruta/al/archivo1 ruta/al/archivo2 ...}}
```

- [c]rea un archivo con formato g[z]ip desde un directorio utilizando rutas relativas:

```
tar czf {{ruta/al/objetivo.tar.gz}} --directory={{ruta/al/directorio}} .
```

- E[x]trae un [f]ichero (comprimido) al directorio actual [v]erbosamente:

```
tar xvf {{ruta/a/fuente.tar[.gz|.bz2|.xz]}}
```

- E[x]trae un [f]ichero (comprimido) al directorio de destino:

```
tar xf {{ruta/al/archivo_de_entrada.tar[.gz|.bz2|.xz]}} --directory={{ruta/al/directorio}}
```

- Crea un archivo comprimido y lo escribe en una carpeta, utilizando la extensión del archivo para determinar automáticamente el programa de compresión:

```
tar caf {{ruta/al/objetivo.tar.xz}} {{ruta/al/archivo1 ruta/al/archivo2 ...}}
```

- Lis[t]a el contenido de un [f]ichero tar [v]erbosamente:

```
tar tvf {{ruta/al/archivo_de_entrada.tar}}
```

- E[x]trae ficheros que coincidan con un patrón de un [f]ichero:

```
tar xf {{ruta/al/archivo_de_entrada.tar}} --wildcards "{{*.html}}"
```



# tee

Lee desde la entrada estándar y escribe a la salida estándar a la vez que a archivos o comandos.

Más información: <https://www.gnu.org/software/coreutils/tee>.

- Copia la entrada estándar al archivo, reemplazando su contenido, y también a la salida estándar:

```
echo {{ejemplo}} | tee {{ruta/al/archivo}}
```

- Anexa la entrada estándar al archivo, sin reemplazar:

```
echo {{ejemplo}} | tee -a {{ruta/al/archivo}}
```

- Imprime la entrada estándar a la terminal, y también lo reenvía a otro programa para posterior procesamiento:

```
echo {{ejemplo}} | tee {{/dev/tty}} | {{xargs printf "[%s]"}}
```

# terraform output

Exporta datos estructurados sobre tus recursos Terraform.

Más información: <https://developer.hashicorp.com/terraform/cli/commands/output>.

- Sin argumentos adicionales, `output` mostrará todas las salidas del módulo raíz:

```
terraform output
```

- Genera solo un valor con un nombre específico:

```
terraform output {{nombre}}
```

- Convierte el valor de salida en una cadena sin formato (útil para scripts de shell):

```
terraform output -raw
```

- Formatea las salidas como un objeto JSON, con una clave por cada salida (útil con jq):

```
terraform output -json
```

# tgpt

Habla con un chatbot de IA sin necesidad de claves API.

Proveedores disponibles: **openai**, **opengpts**, **koboldai**, **phind**, **llama2**, **blackboxai**.

Más información: <https://github.com/aandrew-me/tgpt>.

- Chatea con el proveedor por defecto (GPT-3.5-turbo):

```
tgpt "{{prompt}}"
```

- Inicia el modo interactivo [m]ultilínea:

```
tgpt --multiline
```

- Genera [i]mágenes y las guarda en el directorio actual:

```
tgpt --image "{{prompt}}"
```

- Genera [c]ódigo con el proveedor por defecto (GPT-3.5-turbo):

```
tgpt --code "{{prompt}}"
```

- Chatea con un proveedor específico silenciosamente (sin animaciones):

```
tgpt --provider {{openai|opengpts|koboldai|phind|llama2|blackboxai}} --quiet --whole "{{prompt}}"
```

- Genera y ejecuta comandos de intérprete utilizando un proveedor específico (con confirmación):

```
tgpt --provider {{llama2}} --shell "{{prompt}}"
```

- Pregunta con una clave de API, modelo, longitud máxima de respuesta, temperatura y top\_p (necesario cuando se utiliza el proveedor openai):

```
tgpt --provider openai --key "{{api_key}}" --model  
"{{gpt-3.5-turbo}}" --max-length {{10}} --temperature {{0.7}}  
--top_p {{0.9}} "{{prompt}}"
```

- Alimenta un archivo como pre-entrada adicional:

```
tgpt --provider {{blackboxai}} "{{prompt}}" < {{ruta/al/  
archivo}}
```

# theHarvester

Una herramienta diseñada para las primeras etapas en una prueba de penetración.

Más información: <https://github.com/laramies/theHarvester>.

- Recopila información sobre un dominio utilizando Google:

```
theHarvester --domain {{nombre_de_dominio}} --source google
```

- Recopila información sobre un dominio utilizando varias fuentes:

```
theHarvester --domain {{nombre_de_dominio}} --source  
{{duckduckgo,bing,crtsh}}
```

- Cambia el límite de resultados con los que trabajar:

```
theHarvester --domain {{nombre_de_dominio}} --source  
{{google}} --limit {{200}}
```

- Guarda el resultado en dos archivos en formato XML y HTML:

```
theHarvester --domain {{nombre_de_dominio}} --source  
{{google}} --file {{nombre_de_archivo_de_salida}}
```

- Muestra ayuda:

```
theHarvester --help
```

# tldr

Muestra páginas de ayuda simples para herramientas de línea de comandos del proyecto tldr-pages.

Nota: las opciones **--language** y **--list** no son requeridas por la especificación del cliente, pero la mayoría de los mismos las implementan.

Más información: <https://github.com/tldr-pages/tldr/blob/main/CLIENT-SPECIFICATION.md#command-line-interface>.

- Imprime la página tldr para un comando específico (pista: ¡así es como has llegado hasta aquí!):

```
tldr {{comando}}
```

- Imprime la página tldr para un subcomando específico:

```
tldr {{comando}} {{subcomando}}
```

- Imprime la página tldr para un comando en el [L]enguaje dado (si está disponible, de lo contrario vuelve al inglés):

```
tldr --language {{código_de_lenguaje}} {{comando}}
```

- Imprime la página tldr para un comando de una [p]lataforma específica:

```
tldr --platform {{android|common|freebsd|linux|osx|netbsd|openbsd|sunos|windows}} {{comando}}
```

- Actualiza la caché local de páginas tldr:

```
tldr --update
```

- [l]ista todas las páginas para la plataforma actual y **common**:

```
tldr --list
```

- [l]ista todas las páginas de subcomandos disponibles para un comando:

```
tldr --list | grep {{comando}} | column
```

# tldr

Este comando es un alias de **tldr-lint**.

Más información: <https://github.com/tldr-pages/tldr-lint>.

- Muestra la documentación del comando original:

```
tldr tldr-lint
```

# tlmgr arch

Este comando es un alias de **tlmgr platform**.

Más información: <https://www.tug.org/texlive/tlmgr.html>.

- Muestra la documentación del comando original:

```
tldr tlmgr platform
```

# tmux

Multiplexor de terminal.

Permite múltiples sesiones con ventanas, paneles y más.

Vea también: **zellij**, **screen**.

Más información: <https://github.com/tmux/tmux>.

- Inicia una nueva sesión:

```
tmux
```

- Inicia una nueva sesión con nombre:

```
tmux new -s {{nombre}}
```

- Lista las sesiones existentes:

```
tmux ls
```

- Adjunta a la última sesión utilizada:

```
tmux attach
```

- Separa la sesión actual (dentro de una sesión tmux):

```
<Ctrl>-B d
```

- Crea una nueva ventana (dentro de una sesión tmux):

```
<Ctrl>-B c
```

- Cambia entre sesiones y ventanas (dentro de una sesión tmux):

```
<Ctrl>-B w
```

- Da de baja una sesión por su nombre:

```
tmux kill-session -t {{nombre}}
```



# todoman

Este comando es un alias de **todo**.

Más información: <https://todoman.readthedocs.io/>.

- Muestra la documentación del comando original:

```
tldr todo
```

# toipe

Otra prueba de tipeo, pero con sabor a cangrejo.

Un confiable evaluador de tipeo de terminal.

Más información: <https://github.com/Samyak2/toipe>.

- Inicia el examen de tipeo con la lista de palabras por defecto:

```
toipe
```

- Usa una lista de palabras específica:

```
toipe {{-w|--wordlist}} {{nombre_de_la_lista}}
```

- Utiliza una lista de palabras personalizada:

```
toipe {{-f|--file}} {{ruta/al/archivo}}
```

- Especifica el número de palabras de cada prueba:

```
toipe {{-n|--num}} {{número_de_palabras}}
```

- Incluye signos de puntuación:

```
toipe {{-p|--punctuation}}
```

# touch

Cambia el tiempo de acceso y modificación de un archivo (atime, mtime).

Más información: <https://manned.org/touch>.

- Crea un archivo nuevo o cambia los tiempos de archivos existentes al tiempo actual:

```
touch {{archivo}}
```

- Establece los tiempos de un archivo a un día y hora específicos:

```
touch -t {{YYYYMMDDHHMM.SS}} {{archivo}}
```

- Usa los tiempos de un archivo para establecer los tiempos en otro archivo:

```
touch -r {{archivo}} {{archivo2}}
```

# transmission

Este comando es un alias de **transmission-daemon**.

Más información: <https://transmissionbt.com/>.

- Muestra la documentación del comando original:

```
tldr transmission-daemon
```

# tree

Muestra los contenidos del directorio actual en forma de árbol.

Más información: <https://manned.org/tree>.

- Imprime archivos y directorios hasta `num` niveles de profundidad (donde 1 significa el directorio actual):

```
tree -L {{num}}
```

- Imprime solo directorios:

```
tree -d
```

- Imprime también archivos ocultos, coloreando la salida:

```
tree -a -C
```

- Imprime el árbol sin sangría, mostrando la ruta completa en su lugar (use `-N` para evitar escapar caracteres no imprimibles):

```
tree -i -f
```

- Imprime el tamaño de cada archivo y el tamaño total de cada directorio en formato legible para humanos:

```
tree -s -h --du
```

- Imprime archivos dentro de la jerarquía de árbol, especificando un patrón comodín, excluyendo los directorios que no contengan archivos coincidentes:

```
tree -P '{{*.txt}}' --prune
```

- Imprime archivos dentro de la jerarquía de árbol, especificando un patrón, excluyendo los directorios que no sean ancestros del especificado:

```
tree -P {{nombre_del_directorio}} --matchdirs --prune
```

- Imprime el árbol ignorando los directorios especificados:

```
tree -I '{{nombre_del_directorio1|nombre_del_directorio2}}'
```

# trufflehog

Encuentra y verifica credenciales en archivos, repositorios de Git, cubos S3 e imágenes Docker.

Más información: <https://github.com/trufflesecurity/trufflehog>.

- Escanea un repositorio de Git en busca de secretos verificados:

```
trufflehog git {{https://github.com/trufflesecurity/test_keys}} --only-verified
```

- Escanea una organización de GitHub en busca de secretos verificados:

```
trufflehog github --org={{trufflesecurity}} --only-verified
```

- Escanea un repositorio de GitHub en busca de claves verificadas y genera un archivo de salida JSON:

```
trufflehog git {{https://github.com/trufflesecurity/test_keys}} --only-verified --json
```

- Escanea un repositorio de GitHub junto con sus incidencias y solicitudes de extracción:

```
trufflehog github --repo={{https://github.com/trufflesecurity/test_keys}} --issue-comments --pr-comments
```

- Escanea un cubo S3 en busca de claves verificadas:

```
trufflehog s3 --bucket={{nombre_del_cubo}} --only-verified
```

- Escanea cubos S3 utilizando roles IAM:

```
trufflehog s3 --role-arn={{iam-role-arn}}
```

- Escanea archivos o directorios individuales:

```
trufflehog filesystem {{ruta/al/archivo_o_directorio1 ruta/al/archivo_o_directorio2 ...}}
```

- Escanea una imagen de Docker en busca de secretos verificados:

```
trufflehog docker --image {{trufflesecurity/secrets}} --only-verified
```

# type

Muestra el tipo de comando que ejecutará el intérprete de comandos.

Nota: todos los ejemplos no son compatibles con POSIX.

Más información: <https://manned.org/type>.

- Muestra el tipo de un comando:

```
type {{comando}}
```

- Muestra todas las rutas con el ejecutable especificado (solo funciona en los intérpretes de comandos Bash/fish/Zsh):

```
type -a {{comando}}
```

- Muestra el nombre del archivo en disco que se ejecutaría (solo funciona en intérpretes de comandos Bash/fish/Zsh):

```
type -p {{comando}}
```

- Muestra el tipo de un comando específico, alias/palabra clave/función/integrado/archivo (solo funciona en intérpretes de comandos Bash/fish):

```
type -t {{comando}}
```

# typeinc

Un programa de línea de comandos basado en **ncurses** para probar la velocidad de tecleo, escrito en Python.

Prueba diferentes niveles de dificultad y mejora tu velocidad de tecleo.

Más información: <https://github.com/AnirudhG07/Typeinc>.

- Entra en la prueba de mecanografía:

```
typeinc
```

- Muestra la lista de los 10 primeros clasificados por nivel de dificultad de entrada::

```
typeinc {{-r|--ranklist}} {{nivel_de_dificultad}}
```

- Obtén palabras aleatorias en inglés presentes en nuestra lista de palabras:

```
typeinc {{-w|--words}} {{conteo_de_palabras}}
```

- Calcula el resultado hipotético en Typeinc:

```
typeinc {{-s|--score}}
```



# unlzma

Este comando es un alias de **xz**.

Más información: <https://manned.org/unlzma>.

- Muestra la documentación del comando original:

```
tldr xz
```

# unrar

Extrae archivos RAR.

Más información: <https://manned.org/unrar>.

- Extrae archivos comprimidos respetando la estructura original del archivo:

```
unrar x {{archivo_comprimido.rar}}
```

- Extrae archivos comprimidos en una ruta determinada respetando la estructura original del archivo:

```
unrar x {{archivo_comprimido.rar}} {{ruta/donde/extraer}}
```

- Extrae archivos comprimidos en el directorio actual, perdiendo la estructura original del archivo:

```
unrar e {{archivo_comprimido.rar}}
```

- Comprueba la integridad de cada uno de los archivos dentro del archivo comprimido:

```
unrar t {{archivo_comprimido.rar}}
```

- Muestra el listado de los archivos dentro del archivo comprimido sin descomprimirlo:

```
unrar l {{archivo_comprimido.rar}}
```

# unxz

Este comando es un alias de **xz**.

Más información: <https://manned.org/unxz>.

- Muestra la documentación del comando original:

```
tldr xz
```

# upt

Interfaz unificada para gestionar paquetes en varios sistemas operativos, tales como Windows, diversas distribuciones de Linux, macOS, FreeBSD e incluso Haiku.

Requiere que el gestor de paquetes nativo del sistema operativo esté instalado.

Vea también: **flatpak**, **brew**, **scoop**, **apt**, **dnf**.

Más información: <https://github.com/sigoden/upt>.

- Actualiza la lista de paquetes disponibles:

```
upt update
```

- Busca un paquete:

```
upt search {{término_de_búsqueda}}
```

- Muestra información sobre un paquete:

```
upt info {{paquete}}
```

- Instala un paquete:

```
upt install {{paquete}}
```

- Elimina un paquete:

```
upt {{remove|uninstall}} {{paquete}}
```

- Actualiza todos los paquetes instalados:

```
upt upgrade
```

- Actualiza un paquete:

```
upt upgrade {{paquete}}
```

- Lista los paquetes instalados:

```
upt list
```

# usql

Interfaz de línea de comandos universal para bases de datos SQL.

Más información: <https://github.com/xo/usql>.

- Conecta a una base de datos específica:

```
usql {{sqlserver|mysql|postgres|sqlite3|...}}://  
{{nombre_usuario}}:{{contraseña}}@{{host}}:{{puerto}}/  
{{nombre_base_de_datos}}
```

- Ejecuta comandos desde un archivo:

```
usql --file={{ruta/al/query.sql}}
```

- Ejecuta un comando SQL específico:

```
usql --command="{{comando_sql}}"
```

- Ejecuta un comando SQL en el intérprete de comandos de usql:

```
{{prompt}}=> {{comando}}
```

- Muestra el esquema de la base de datos:

```
{{prompt}}=> \d
```

- Exporta los resultados de la consulta a un archivo específico:

```
{{prompt}}=> \g {{ruta/a/archivo_con_resultados}}
```

- Importa datos de un archivo CSV a una tabla específica:

```
{{prompt}}=> \copy {{ruta/a/datos.csv}} {{nombre_de_tabla}}
```

# uv python

Gestiona las versiones e instalaciones de Python.

Más información: <https://docs.astral.sh/uv/reference/cli/#uv-python>.

- Lista todas las instalaciones de Python disponibles:

```
uv python list
```

- Instala una versión de Python:

```
uv python install {{versión}}
```

- Desinstala una versión de Python:

```
uv python uninstall {{version}}
```

- Busca una instalación de Python:

```
uv python find {{versión}}
```

- Fija el proyecto actual para utilizar una versión específica de Python:

```
uv python pin {{versión}}
```

- Muestra el directorio de instalación de uv Python:

```
uv python dir
```

# vboxmanage movevm

Mueve una máquina virtual (VM) a una nueva ubicación en el sistema anfitrión.

Más información: <https://www.virtualbox.org/manual/ch08.html#vboxmanage-movevm>.

- Mueve la máquina virtual especificada a la ubicación actual:

```
VBoxManage movevm {{nombre_vm}}
```

- Especifica la nueva ubicación (nombre de ruta completo o relativo) de la máquina virtual:

```
VBoxManage movevm {{nombre_vm}} --folder {{ruta/a/  
nueva_ubicación}}
```

# vcpkg

Gestor de paquetes para librerías C/C++.

Nota: los paquetes no se instalan en el sistema. Para usarlos, necesitas decirle a tu sistema de compilación (por ejemplo CMake) que use **vcpkg**.

Más información: <https://learn.microsoft.com/en-us/vcpkg/>.

- Construye y añade el paquete `libcurl` al entorno de `vcpkg`:

```
vcpkg install curl
```

- Construye y añade `zlib` usando la cadena de herramientas `emscripten`:

```
vcpkg install --triplet=wasm32-emscripten zlib
```

- Busca un paquete:

```
vcpkg search {{nombre_del_paquete}}
```

- Configura un proyecto CMake para utilizar los paquetes de `vcpkg`:

```
cmake -B build -DCMAKE_TOOLCHAIN_FILE={{ruta/al/  
directorio_de_instalación_vcpkg}}/scripts/buildsystems/  
vcpkg.cmake
```



# vi

Este comando es un alias de **vim**.

- Muestra la documentación del comando original:

```
tldr vim
```

# vim

Vim (Vi IMproved), un editor de texto para la línea de comandos, que proporciona varios modos para diferentes tipos de manipulación de texto.

Pulsando **i** entra en el modo insertar. **<Esc>** regresa al modo normal, permitiendo el uso de comandos Vim.

Más información: <https://www.vim.org>.

- Abre un archivo:

```
vim {{ruta/al/archivo}}
```

- Abre un archivo en un número de línea especificado:

```
vim +{{número_de_línea}} {{ruta/al/archivo}}
```

- Muestra el manual de Vim:

```
:help<Enter>
```

- Guarda y sale:

```
{{ZZ| :wq<Enter>}}
```

- Deshaz la última operación:

```
<ESC>u
```

- Busca un patrón en el archivo (pulsas n/N para ir a la próxima/previa coincidencia):

```
/{{patrón_a_buscar}}<Enter>
```

- Realiza una sustitución de una expresión regular en el archivo completo:

```
:%s/{{expresión_regular}}/{{reemplazo}}/g<Enter>
```

- Muestra los números de línea:

```
:set nu<Enter>
```

# vimdiff

Abre dos o más archivos en Vim y muestra las diferencias entre ellos.

Vea también **vim**.

Más información: <https://www.vim.org>.

- Abre dos archivos y muestra las diferencias:

```
vimdiff {{archivo1}} {{archivo2}}
```

- Mueve el cursor a la ventana de la izquierda | derecha:

```
<Ctrl> + w {{h|l}}
```

- Salta a la diferencia previa:

```
[c
```

- Salta a la siguiente diferencia:

```
]c
```

- Copia la diferencia resaltada de la otra ventana a la ventana actual:

```
do
```

- Copia la diferencia resaltada de la ventana actual a la otra ventana:

```
dp
```

- Actualiza todos los resaltados y folds (plegados de texto):

```
:diffupdate
```

- Alterna la apertura/cierre de la fold (plegado de texto) de código resaltada:

```
za
```

# vimtutor

Vim tutor, enseña los comandos básicos de vim.

Más información: <https://manned.org/vimtutor>.

- Ejecuta vim tutor utilizando el idioma especificado (en, es, de, ...):

```
vimtutor {{idioma}}
```

- Sale del tutor:

```
<Esc> :q <Enter>
```

# virt-qemu-run

Herramienta experimental para ejecutar una Guest VM QEMU independiente de **libvirtd**.

Más información: <https://libvirt.org/manpages/virt-qemu-run.html>.

- Ejecuta una máquina virtual QEMU:

```
virt-qemu-run {{ruta/a/guest.xml}}
```

- Ejecuta una máquina virtual QEMU y almacena el estado en un directorio específico:

```
virt-qemu-run --root={{ruta/a/directorio}} {{ruta/a/guest.xml}}
```

- Ejecuta una máquina virtual QEMU y muestra información detallada sobre el inicio:

```
virt-qemu-run --verbose {{ruta/a/guest.xml}}
```

- Muestra ayuda:

```
virt-qemu-run --help
```

# Vite

Crea un proyecto Vite.

Se utiliza para crear proyectos JavaScript.

Plantillas disponibles: vanilla, vanilla-ts, vue, vue-ts, react, react-ts, react-swc, react-swc-ts, preact, preact-ts, lit, lit-ts, svelte, svelte-ts.

Más información: <https://vitejs.dev/guide>.

- Configuración usando npm 6.x:

```
npm create vite@latest my-react-app --template react-ts
```

- Configuración usando npm 7+, se necesita un guión doble adicional:

```
npm create vite@latest my-react-app -- --template react-ts
```

- Configuración usando yarn:

```
yarn create vite my-react-app --template react-ts
```

- Configuración usando pnpm:

```
pnpm create vite my-react-app --template react-ts
```

# vulkaninfo

Imprime información del sistema Vulkan.

Más información: <https://vulkan.lunarg.com/doc/view/latest/linux/vulkaninfo.html>.

- Imprime la información completa de Vulkan:

```
vulkaninfo
```

- Imprime un resumen:

```
vulkaninfo --summary
```

- Genera un documento HTML con la información completa de Vulkan:

```
vulkaninfo --html
```

# wafw00f

Identifica y toma las huellas digitales de los productos Web Application Firewall (WAF) que protegen un sitio web.

Más información: <https://github.com/EnableSecurity/wafw00f>.

- Comprueba si un sitio web utiliza algún WAF:

```
wafw00f {{https://www.ejemplo.com}}
```

- Comprueba a todos los WAF detectables sin detenerse en la primera coincidencia:

```
wafw00f --findall {{https://www.ejemplo.com}}
```

- Pasa peticiones a través de un [p]roxy (como BurpSuite):

```
wafw00f --proxy {{http://localhost:8080}} {{https://  
www.ejemplo.com}}
```

- [t]estea un producto WAF específico (ejecuta `wafw00f -l` para obtener una lista de todos los WAF compatibles):

```
wafw00f --test {{Cloudflare|Cloudfront|Fastly|ZScaler|...}}  
{{https://www.ejemplo.com}}
```

- Pasa cabeceras personalizadas desde un archivo:

```
wafw00f --headers {{ruta/a/cabeceras.txt}} {{https://  
www.ejemplo.com}}
```

- Lee entradas de destino desde un archivo y muestra una salida detallada (múltiples `v` para más verbosidad):

```
wafw00f --input {{ruta/a/urls.txt}} -v{{v}}
```

- [l]ista todos los WAF que pueden detectarse:

```
wafw00f --list
```



# wakeonlan

Envía paquetes a los PCs habilitados para wake-on-LAN (WOL).

Más información: <https://github.com/jpoliv/wakeonlan>.

- Envía paquetes a todos los dispositivos de la red local (255.255.255.255) especificando una dirección MAC:

```
wakeonlan {{01:02:03:04:05:06}}
```

- Envía paquete a un dispositivo específico a través de una dirección IP:

```
wakeonlan {{01:02:03:04:05:06}} -i {{192.168.178.2}}
```

- Imprime los comandos, pero no los ejecutes (dry-run):

```
wakeonlan -n {{01:02:03:04:05:06}}
```

- Ejecuta en modo silencioso:

```
wakeonlan -q {{01:02:03:04:05:06}}
```

# waybar

Barra Wayland altamente personalizable para compositores basados en Sway y Wlroots.

Más información: <https://github.com/Alexays/Waybar>.

- Inicia `waybar` con la configuración y hoja de estilos predeterminada:

```
waybar
```

- Usa un archivo de configuración diferente:

```
waybar {{-c|--config}} {{ruta/a/  
archivo_de_configuración.jsonc}}
```

- Utiliza un archivo de hoja de estilo diferente:

```
waybar {{-s|--style}} {{ruta/a/hoja_de_estilo.css}}
```

- Establece el nivel de registro:

```
waybar {{-l|--log-level}} {{trace|debug|info|warning|error|  
critical|off}}
```

# waymore

Obtén las URLs de un dominio desde Wayback Machine, Common Crawl, Alien Vault OTX, URLScan y VirusTotal.

Nota: A menos que se especifique lo contrario, los resultados se escriben en el directorio **results/** donde reside el archivo **config.yml** de waymore (por defecto en **~/.config/waymore/**).

Más información: <https://github.com/xnl-h4ck3r/waymore>.

- Busca URLs de un dominio (la salida suele estar en **~/.config/waymore/results/**):

```
waymore -i {{ejemplo.com}}
```

- Limita los resultados de la búsqueda para que solo incluyan una lista de URLs de un dominio y almacena los resultados en el archivo especificado:

```
waymore -mode U -oU {{ruta/a/ejemplo.com-urls.txt}} -i {{ejemplo.com}}
```

- Imprime solo los cuerpos de contenido de las URLs y almacena los resultados en el directorio especificado:

```
waymore -mode R -oR {{ruta/a/ejemplo.com-url-responses}} -i {{ejemplo.com}}
```

- Filtra los resultados especificando intervalos de fechas:

```
waymore -from {{YYYYMMDD|YYMM|YYYY}} -to {{AAAAMMDD|AAAAMMD|AAAAMMD}} -i {{ejemplo.com}}
```

# WC

Cuenta líneas, palabras, y bytes.

Más información: <https://www.gnu.org/software/coreutils/wc>.

- Cuenta todas las líneas en un archivo:

```
wc --lines {{ruta/al/archivo}}
```

- Cuenta todas las palabras en un archivo:

```
wc --words {{ruta/al/archivo}}
```

- Cuenta todos los bytes en un archivo:

```
wc --bytes {{ruta/al/archivo}}
```

- Cuenta todos los caracteres en un archivo (considerando los caracteres de varios bytes):

```
wc --chars {{ruta/al/archivo}}
```

- Cuenta todas las líneas, palabras y bytes desde `stdin`:

```
{{find .}} | wc
```

- Cuenta la longitud de la línea más larga en número de caracteres:

```
wc --max-line-length {{ruta/al/archivo}}
```

# wdiff

Muestra las diferencias de palabras entre archivos de texto.

Más información: <https://www.gnu.org/software/wdiff/>.

- Compara dos archivos:

```
wdiff {{ruta/al/archivo1}} {{ruta/al/archivo2}}
```

- Ignora mayúsculas y minúsculas al comparar:

```
wdiff --ignore-case {{ruta/al/archivo1}} {{ruta/al/archivo2}}
```

- Muestra cuantas palabras se han eliminado, insertado o sustituido:

```
wdiff --statistics {{ruta/al/archivo1}} {{ruta/al/archivo2}}
```

# whatwaf

Detecta y elude cortafuegos de aplicaciones web y sistemas de protección.

Más información: <https://github.com/Ekultek/WhatWaf>.

- Detecta protección en una sola [u]RL, opcionalmente utiliza salida verbose:

```
whatwaf --url {{https://example.com}} --verbose
```

- Detecta protección en un [l]ista de URLs en paralelo desde un archivo (una URL por línea):

```
whatwaf --threads {{número}} --list {{ruta/a/archivo}}
```

- Envía peticiones a través de un proxy y utiliza una lista de carga útil personalizada desde un archivo (una carga útil por línea):

```
whatwaf --proxy {{http://127.0.0.1:8080}} --pl {{ruta/a/archivo}} -u {{https://example.com}}
```

- Envía peticiones a través de Tor (Tor debe estar instalado) utilizando cargas personalizadas (separadas por comas):

```
whatwaf --tor --payloads '{{carga1,carga2,...}}' -u {{https://example.com}}
```

- Utiliza un agente de usuario aleatorio, establece el estrangulamiento y el tiempo de espera, envía una solicitud [P]OST y fuerza una conexión HTTPS:

```
whatwaf --ra --throttle {{segundos}} --timeout {{segundos}} --post --force-ssl -u {{http://example.com}}
```

- Enumera todos los WAF que se pueden detectar:

```
whatwaf --wafs
```

- Lista todos los scripts de manipulación disponibles:

```
whatwaf --tamperers
```

# xkill

Cierra de manera forzosa una ventana interactivamente en una sesión gráfica.

Vea también **kill** y **killall**.

Más información: <https://www.x.org/releases/current/doc/man/man1/xkill.1.xhtml>.

- Muestra un cursor para cerrar forzosamente una ventana presionando con el botón izquierdo (presiona cualquier otro botón del ratón para cancelar):

```
xkill
```

- Muestra un cursor para seleccionar una ventana para cerrarla forzosamente al presionar cualquier botón del ratón:

```
xkill -button any
```

- Cierra forzosamente una ventana con un ID específico (use `xwininfo` para obtener información acerca de las ventanas):

```
xkill -id {{id}}
```

# xmake

Una utilidad de compilación multiplataforma C & C++ basada en Lua.

Más información: [https://xmake.io/#/getting\\_started](https://xmake.io/#/getting_started).

- Crea un proyecto Xmake C, consistente en un hello world y `xmake.lua`:

```
xmake create --language c -P {{nombre_del_proyecto}}
```

- Construye y ejecuta un proyecto Xmake:

```
xmake build run
```

- Ejecuta directamente un objetivo Xmake compilado:

```
xmake run {{nombre_del_objetivo}}
```

- Configura los objetivos de compilación de un proyecto:

```
xmake config --plat={{macosx|linux|iphoneos|...}} --  
arch={{x86_64|i386|arm64|..}} --mode={{debug|release}}
```

- Instala el objetivo compilado en un directorio:

```
xmake install -o {{ruta/al/directorio}}
```



# xzcat

Este comando es un alias de **xz**.

Más información: <https://manned.org/xzcat>.

- Muestra la documentación del comando original:

```
tldr xz
```

# ya

Gestiona los paquetes y complementos de Yazi.

Más información: <https://github.com/sxyazi/yazi>.

- Añade un paquete:

```
ya pack -a {{paquete}}
```

- Actualiza todos los paquetes:

```
ya pack -u
```

- Suscribirse a los mensajes de todas las instancias remotas:

```
ya sub {{tipos}}
```

- Publica un mensaje en la instancia actual con cuerpo de cadena:

```
ya pub --str {{cadena_mensaje}}
```

- Publica un mensaje a la instancia actual con formato JSON:

```
ya pub --json {{mensaje_json}}
```

- Publica un mensaje en la instancia especificada con una cadena de texto:

```
ya pub-to --str {{mensaje}} {{receptor}} {{tipo}}
```

# yacc

Genera un analizador sintáctico LALR (en C) con un archivo de especificación de gramática formal.

Vea también: **bison**.

Más información: <https://manned.org/yacc.1p>.

- Crea un fichero `y.tab.c` con el código del analizador en C y compila el fichero de gramática con todas las declaraciones constantes necesarias para los valores. El fichero `y.tab.h` con las declaraciones se crea exclusivamente con la opción `-d`:

```
yacc -d {{ruta/al/archivo_de_gramática.y}}
```

- Compila un fichero de gramática con la descripción del analizador sintáctico y un informe de conflictos generados por ambigüedades en la gramática:

```
yacc -d {{ruta/al/archivo_de_gramática.y}} -v
```

- Compila un archivo de gramática, y prefija los nombres de los archivos de salida con un prefijo personalizado en lugar de `y`:

```
yacc -d {{ruta/al/archivo_de_gramática.y}} -v -b {{prefijo}}
```

# yazi

Rápido gestor de archivos escrito en Rust.

Experiencia de gestión de archivos eficiente, fácil de usar y personalizable.

Más información: <https://github.com/sxyazi/yazi>.

- Inicia Yazi desde el directorio actual:

```
yazi
```

- Imprime información de depuración:

```
yazi --debug
```

- Escribe el directorio de trabajo actual al salir del archivo:

```
yazi --cwd-file {{ruta/a/archivo_cwd}}
```

- Borra el directorio de caché:

```
yazi --clear-cache
```

# yes

Retorna algo repetidamente.

Este comando es frecuentemente utilizado para aceptar todas las confirmaciones en comandos de instalación (como apt-get).

Más información: <https://www.gnu.org/software/coreutils/yes>.

- Retorna repetidamente "mensaje":

```
yes {{mensaje}}
```

- Retorna repetidamente "y":

```
yes
```

- Acepta todas las confirmaciones que muestre el comando apt-get:

```
yes | sudo apt-get install {{programa}}
```

# yuy2topam

Convierte bytes YUY2 a PAM.

Más información: <https://netpbm.sourceforge.net/doc/yuy2topam.html>.

- Convierte bytes YUY2 a PAM:

```
yuy2topam -width {{valor}} -height {{valor}} {{ruta/al/  
archivo.yuy2}} > {{ruta/al/archivo.pam}}
```

# zint

Genera códigos de barras y códigos QR.

Más información: <https://www.zint.org.uk/manual/chapter/4>.

- Crea un archivo con un código de barras:

```
zint --data "{{datos_UTF-8}}" --output {{ruta/al/archivo}}
```

- Crea un archivo con otro tipo de código de barras:

```
zint --barcode {{tipo_de_código}} --data "{{datos_UTF-8}}" --  
output {{ruta/al/archivo}}
```

- Lista todos los tipos de códigos de barras soportados:

```
zint --types
```

# zipcloak

Encripta el contenido de un archivo zip.

Más información: <https://manned.org/zipcloak>.

- Encripta el contenido de un archivo zip:

```
zipcloak {{ruta/al/archivo.zip}}
```

- [d]esencripta el contenido de un archivo zip:

```
zipcloak -d {{ruta/al/archivo_encriptado.zip}}
```

- Genera un nuev[O] archivo zip con el contenido encriptado:

```
zipcloak {{ruta/al/archivo.zip}} -O {{ruta/al/archivo_encriptado.zip}}
```



# zsh

Z SHell, un intérprete de línea de comandos compatible con Bash.

Vea también **histexpand** para la expansión del historial.

Más información: <https://www.zsh.org>.

- Comienza una sesión interactiva en el intérprete de comandos:

```
zsh
```

- Ejecuta un comando y sale:

```
zsh -c "{{comando}}"
```

- Ejecuta un script:

```
zsh {{ruta/al/script.zsh}}
```

- Comprueba si hay errores de sintaxis en un script sin ejecutarlo:

```
zsh --no-exec {{ruta/al/script.zsh}}
```

- Ejecuta comandos desde stdin:

```
{{comando}} | zsh
```

- Ejecuta un script, mostrando cada comando antes de ejecutarlo:

```
zsh --xtrace {{ruta/al/script.zsh}}
```

- Comienza una sesión interactiva en el intérprete de comandos en modo detallado, mostrando cada comando antes de ejecutarlo:

```
zsh --verbose
```

- Ejecuta un comando dentro de `zsh` con patrones glob desactivados:

```
noglob {{command}}
```

Freebsd

# look

Muestra las líneas que empiezan con un prefijo en un archivo ordenado.

Vea también: **grep**, **sort**.

Más información: <https://man.freebsd.org/cgi/man.cgi?look>.

- Busca líneas que comiencen con un prefijo específico en un archivo específico:

```
look {{prefijo}} {{ruta/al/archivo}}
```

- Busca caracteres alfanuméricos sin tomar en cuenta las mayúsculas o minúsculas:

```
look {{-f|--ignore-case}} {{-d|--alphanum}} {{prefijo}}  
{{ruta/al/archivo}}
```

- Especifica un carácter de terminación de cadena (un espacio por defecto):

```
look {{-t|--terminate}} {{,}}
```

- Busca en /usr/share/dict/words (se asumen --ignore-case y --alphanum):

```
look {{prefijo}}
```

# pkg

Gestor de paquetes de FreeBSD.

Más información: <https://man.freebsd.org/cgi/man.cgi?pkg>.

- Instala un nuevo paquete:

```
pkg install {{paquete}}
```

- Elimina un paquete:

```
pkg delete {{paquete}}
```

- Actualiza todos los paquetes:

```
pkg upgrade
```

- Busca un paquete:

```
pkg search {{palabra_clave}}
```

- Lista los paquetes instalados:

```
pkg info
```

- Elimina dependencias innecesarias:

```
pkg autoremove
```

Linux

# a2disconf

Deshabilita un archivo de configuración de Apache en sistemas operativos basados en Debian.

Más información: <https://manned.org/a2disconf.8>.

- Deshabilita un archivo de configuración:

```
sudo a2disconf {{archivo_de_configuración}}
```

- Deshabilita un archivo sin mostrar mensajes informativos:

```
sudo a2disconf --quiet {{archivo_de_configuración}}
```

# a2dismod

Deshabilita un módulo de Apache en sistemas operativos basados en Debian.

Más información: <https://manned.org/a2dismod.8>.

- Deshabilita un módulo:

```
sudo a2dismod {{módulo}}
```

- Deshabilita un módulo sin mostrar mensajes informativos:

```
sudo a2dismod --quiet {{módulo}}
```

# a2dissite

Deshabilita un servidor virtual Apache en sistemas operativos basados en Debian.

Más información: <https://manned.org/a2dissite.8>.

- Deshabilita un host virtual:

```
sudo a2dissite {{host_virtual}}
```

- Deshabilita un host virtual sin mostrar mensajes informativos:

```
sudo a2dissite --quiet {{host_virtual}}
```



# a2enconf

Habilita un archivo de configuración de Apache en sistemas operativos basados en Debian.

Más información: <https://manned.org/a2enconf.8>.

- Habilita un archivo de configuración:

```
sudo a2enconf {{archivo_de_configuración}}
```

- Habilita un archivo de configuración sin mostrar mensajes informativos:

```
sudo a2enconf --quiet {{archivo_de_configuración}}
```

# a2enmod

Habilita un módulo de Apache en sistemas operativos basados en Debian.

Más información: <https://manned.org/a2enmod.8>.

- Habilita un módulo:

```
sudo a2enmod {{módulo}}
```

- Habilita un módulo sin mostrar mensajes informativos:

```
sudo a2enmod --quiet {{módulo}}
```

# a2ensite

Habilita un servidor virtual Apache en sistemas operativos basados en Debian.

Más información: <https://manned.org/a2ensite.8>.

- Habilita un host virtual:

```
sudo a2ensite {{host_virtual}}
```

- Habilita un host virtual sin mostrar mensajes informativos:

```
sudo a2ensite --quiet {{host_virtual}}
```

# a2query

Recupera la configuración en tiempo de ejecución de Apache en sistemas operativos basados en Debian.

Más información: <https://manned.org/a2query>.

- Lista módulos de Apache habilitados:

```
sudo a2query -m
```

- Comprueba si un módulo específico está instalado:

```
sudo a2query -m {{nombre_del_módulo}}
```

- Lista hosts virtuales habilitados:

```
sudo a2query -s
```

- Muestra el Módulo de Procesamiento Múltiple actualmente habilitado:

```
sudo a2query -M
```

- Muestra la versión de Apache:

```
sudo a2query -v
```

# abbr

Administra abreviaturas del intérprete de comandos fish.

Las palabras definidas por el usuario se reemplazan con frases más largas después de ingresarlas.

Más información: <https://fishshell.com/docs/current/cmds/abbr.html>.

- Añade una nueva abreviatura:

```
abbr --add {{nombre_abreviatura}} {{comando}}  
{{argumentos_del_comando}}
```

- Cambia el nombre de una abreviatura existente:

```
abbr --rename {{nombre_antiguo}} {{nombre_nuevo}}
```

- Borra una abreviatura existente:

```
abbr --erase {{nombre_abreviatura}}
```

- Importa las abreviaturas definidas en otro host a través de SSH:

```
ssh {{nombre_host}} abbr --show | source
```

# abroot

Utilidad que proporciona completa inmutabilidad y atomicidad mediante transacciones entre 2 estados de partición de la raíz ( $A \rightleftharpoons B$ ).

Las actualizaciones se realizan utilizando imágenes OCI, para asegurar que el sistema está siempre en un estado consistente.

Más información: <https://github.com/Vanilla-OS/ABRoot>.

- Añade paquetes a la imagen local (Nota: después de ejecutar este comando, se necesita aplicar estos cambios):

```
sudo abroot pkg add {{paquete}}
```

- Elimina paquetes de la imagen local (Nota: después de ejecutar este comando, debe aplicar estos cambios):

```
sudo abroot pkg remove {{paquete}}
```

- Lista paquetes en la imagen local:

```
sudo abroot pkg list
```

- Aplica los cambios en la imagen local (Nota: es necesario reiniciar el sistema para que estos cambios sean aplicados):

```
sudo abroot pkg apply
```

- Retrocede su sistema al estado anterior:

```
sudo abroot rollback
```

- Edita/Visualiza los parámetros del kernel:

```
sudo abroot kargs {{edit|show}}
```

- Muestra estado:

```
sudo abroot status
```

- Muestra ayuda:

```
abroot --help
```

# ac

Imprime estadísticas sobre cuánto tiempo han estado conectados los usuarios.

Más información: <https://www.gnu.org/software/acct/manual/accounting.html#ac>.

- Imprime cuánto tiempo ha estado conectado el usuario actual en horas:

```
ac
```

- Imprime cuánto tiempo han estado conectados los usuarios en horas:

```
ac --individual-totals
```

- Imprime cuánto tiempo ha estado conectado un usuario en particular en horas:

```
ac --individual-totals {{nombre_usuario}}
```

- Imprime cuánto tiempo un usuario en particular ha estado conectado en horas por día (en total):

```
ac --daily-totals --individual-totals {{nombre_usuario}}
```

- Muestra también detalles adicionales:

```
ac --compatibility
```

# acountry

Imprime el país donde se encuentra una dirección IPv4 o el nombre de un host.

Más información: <https://manned.org/acountry>.

- Imprime un país donde se encuentra una dirección IPv4 o host:

```
acountry {{ejemplo.com}}
```

- Imprime salida de depuración adicional:

```
acountry -d {{ejemplo.com}}
```

- Imprime información más detallada:

```
acountry -v {{ejemplo.com}}
```



# agetty

Alternativa a **getty**: Abre un puerto **tty**, pide un nombre de usuario, e invoca el comando **/bin/login**.

Normalmente es invocado por **init**.

Nota: la tasa de baudios es la velocidad de transferencia de datos entre una terminal y un dispositivo a través de una conexión serie.

Más información: <https://manned.org/agetty>.

- Conecta **stdin** a un puerto (relativo a **/dev**) y especifica opcionalmente una tasa de baudios (cuyo valor predeterminado es 9600):

```
agetty {{tty}} {{115200}}
```

- Asume que **stdin** ya está conectado a una **tty** y establece un tiempo de espera para el inicio de sesión:

```
agetty {{-t|--timeout}} {{tiempo_de_espera_en_segundos}} -
```

- Asume que **tty** es de [8]-bits, sobrescribiendo la variable de entorno **TERM** establecida por **init**:

```
agetty -8 - {{variable_term}}
```

- Omite el inicio de sesión e invoca, como superusuario, otro programa de inicio de sesión en lugar de **/bin/login**:

```
agetty {{-n|--skip-login}} {{-l|--login-program}}  
{{programa_de_inicio_de_sesión}} {{tty}}
```

- Escribe el mensaje de inicio de sesión sin mostrar el contenido del archivo de pre-inicio de sesión (**/etc/issue** por predeterminado):

```
agetty {{-i|--noissue}} -
```

- Cambia el directorio raíz y escribe un host falso en el archivo **utmp**:

```
agetty {{-r|--chroot}} {{/ruta/a/raíz_directorio}} {{-H|--  
host}} {{host_falso}} -
```

# alternatives

Este comando es un alias de **update-alternatives**.

Más información: <https://manned.org/alternatives>.

- Muestra la documentación del comando original:

```
tldr update-alternatives
```

# apk

Herramienta de gestión de paquetes de Alpine Linux.

Más información: <https://manned.org/apk>.

- Actualiza los índices de repositorio desde todos los repositorios remotos:

```
apk update
```

- Instala un nuevo paquete:

```
apk add {{paquete}}
```

- Remueve un paquete:

```
apk del {{paquete}}
```

- Repara un paquete o lo actualiza sin modificar dependencias principales:

```
apk fix {{paquete}}
```

- Busca un paquete usando palabras clave:

```
apk search {{palabras_clave}}
```

- Muestra información sobre un paquete específico:

```
apk info {{paquete}}
```

# apt-add-repository

Gestiona las definiciones del repositorio APT.

Más información: <https://manned.org/apt-add-repository.1>.

- Añade un nuevo repositorio APT:

```
apt-add-repository {{repositorio}}
```

- Elimina un repositorio APT:

```
apt-add-repository --remove {{repositorio}}
```

- Actualiza la caché de paquetes tras añadir un repositorio:

```
apt-add-repository --update {{repositorio}}
```

- Activa las fuentes de paquetes:

```
apt-add-repository --enable-source {{repositorio}}
```

# apt-cache

Herramienta de consulta de paquetes para Debian y Ubuntu.

Más información: <https://manned.org/apt-cache.8>.

- Busca un paquete en tus fuentes actuales:

```
apt-cache search {{consulta}}
```

- Muestra información de un paquete:

```
apt-cache show {{paquete}}
```

- Muestra si un paquete está instalado y actualizado:

```
apt-cache policy {{paquete}}
```

- Muestra las dependencias de un paquete:

```
apt-cache depends {{paquete}}
```

- Muestra los paquetes que dependen de un paquete en particular:

```
apt-cache rdepends {{paquete}}
```

# apt-file

Busca archivos en paquetes APT, incluyendo los que aún no fueron instalados.

Más información: <https://manned.org/apt-file.1>.

- Actualiza los metadatos de la base de datos:

```
sudo apt update
```

- Busca paquetes que contengan el archivo o ruta especificada:

```
apt-file search {{ruta/al/archivo}}
```

- Muestra el contenido del paquete especificado:

```
apt-file list {{nombre_paquete}}
```

# apt-get

Utilidad de gestión de paquetes para Debian y Ubuntu.

Búsqueda de paquetes mediante **apt-cache**.

Más información: <https://manned.org/apt-get.8>.

- Actualiza la lista de paquetes y versiones disponibles (se recomienda ejecutar esto antes de otros comandos `apt-get`):

```
apt-get update
```

- Instala un paquete o lo actualiza a la última versión disponible:

```
apt-get install {{paquete}}
```

- Elimina un paquete:

```
apt-get remove {{paquete}}
```

- Elimina un paquete y sus archivos de configuración:

```
apt-get purge {{paquete}}
```

- Actualiza todos los paquetes instalados a sus versiones más recientes:

```
apt-get upgrade
```

- Limpia el repositorio local: elimina los archivos de paquetes (.deb) de descargas interrumpidas que ya no pueden descargarse:

```
apt-get autoclean
```

- Elimina todos los paquetes que ya no sean necesarios:

```
apt-get autoremove
```

- Actualiza los paquetes instalados (como `upgrade`), pero eliminando los paquetes obsoletos e instalando paquetes adicionales para satisfacer las nuevas dependencias:

```
apt-get dist-upgrade
```

# apt-key

Herramienta para la gestión de claves para el Gestor de Paquetes APT (APT Package Manager) en Debian y Ubuntu.

Más información: <https://manned.org/apt-key.8>.

- Muestra las claves de confianza:

```
apt-key list
```

- Añade una clave al almacén de claves de confianza:

```
apt-key add {{archivo_clave_pública.asc}}
```

- Borra una clave del almacén de claves de confianza:

```
apt-key del {{identificador_de_clave}}
```

- Añade una clave remota al almacén de claves de confianza:

```
wget -q0 - {{https://host.tld/archivo.clave}} | apt-key add -
```

- Añade una clave de un servidor de claves con el identificador de la clave:

```
apt-key adv --keyserver {{pgp.mit.edu}} --recv  
{{identificador_de_clave}}
```



# apt-mark

Herramienta para cambiar el estado de los paquetes instalados.

Más información: <https://manned.org/apt-mark.8>.

- Marca un paquete como instalado automáticamente:

```
sudo apt-mark auto {{nombre_paquete}}
```

- Mantiene un paquete en su versión actual y evita que se actualice:

```
sudo apt-mark hold {{nombre_paquete}}
```

- Permite que un paquete pueda ser actualizado de nuevo:

```
sudo apt-mark unhold {{nombre_paquete}}
```

- Muestra los paquetes instalados manualmente:

```
apt-mark showmanual
```

- Muestra los paquetes mantenidos que no son actualizados:

```
apt-mark showhold
```

# apt

Herramienta de gestión de paquete para distribuciones basadas en Debian.

Se recomienda sustituirlo por **apt-get** cuando se use interactivamente en Ubuntu 16.04 o versiones posteriores.

Más información: <https://manned.org/apt.8>.

- Actualiza la lista de paquetes y versiones disponibles (se recomienda ejecutar este comando antes que cualquier otro comando **apt**):

```
sudo apt update
```

- Busca un paquete:

```
apt search {{paquete}}
```

- Muestra la información de un paquete:

```
apt show {{paquete}}
```

- Instala un paquete o lo actualiza a su última versión disponible:

```
sudo apt install {{paquete}}
```

- Elimina un paquete (si se utiliza **purge** también elimina sus archivos de configuración):

```
sudo apt remove {{paquete}}
```

- Actualiza todos los paquetes a sus nuevas versiones disponibles:

```
sudo apt upgrade
```

- Muestra todos los paquetes:

```
apt list
```

- Muestra los paquetes instalados:

```
apt list --installed
```

# aptitude

Herramienta de gestión de paquetes para Debian y Ubuntu.

Más información: <https://manned.org/aptitude.8>.

- Sincroniza la lista de paquetes y versiones disponible (se recomienda ejecutar este comando antes que cualquier otro comando `aptitude`):

```
aptitude update
```

- Instala un nuevo paquete y sus dependencias:

```
aptitude install {{paquete}}
```

- Busca un paquete:

```
aptitude search {{paquete}}
```

- Busca un paquete instalado (?installed es un término de búsqueda de `aptitude`):

```
aptitude search '?installed({{paquete}})'
```

- Elimina un paquete y todos los paquetes que dependen de él:

```
aptitude remove {{paquete}}
```

- Actualiza todos los paquetes a sus nuevas versiones disponibles:

```
aptitude upgrade
```

- Actualiza paquetes instalados (como `aptitude upgrade`), elimina los paquetes obsoletos e instala paquetes adicionales para satisfacer sus dependencias:

```
aptitude full-upgrade
```

- Mantiene un paquete instalado para que no sea actualizado automáticamente:

```
aptitude hold '?installed({{paquete}})'
```

# archey

Herramienta sencilla para mostrar información del sistema con estilo.

Más información: <https://lclarkmichalek.github.io/archey3/>.

- Muestra información del sistema:

archey

# archinstall

Instalador guiado de Arch Linux con un giro.

Más información: <https://archinstall.readthedocs.io>.

- Inicia el instalador interactivo:

```
archinstall
```

- Inicia un instalador preestablecido:

```
archinstall {{minimal|unattended}}
```

# audit2allow

Crea un módulo de política local SELinux para permitir reglas basadas en operaciones denegadas encontradas en los logs.

Nota: Utiliza audit2allow con precaución - revisa siempre la directiva generada antes de aplicarla, ya que puede permitir un acceso excesivo.

Más información: <https://manned.org/audit2allow>.

- Genera una política local para permitir el acceso a todos los servicios denegados:

```
sudo audit2allow --all -M {{nombre_de_la_normativa_local}}
```

- Genera un módulo de normativa local para conceder acceso a un proceso/servicio/comando específico de los registros de auditoría:

```
sudo grep {{apache2}} /var/log/audit/audit.log | sudo  
audit2allow -M {{nombre_de_la_normativa_local}}
```

- Inspecciona y revisa el archivo Type Enforcement (.te) para una normativa local:

```
vim {{nombre_de_la_normativa_local}}.te
```

- Instala un módulo de normativa local:

```
sudo semodule -i {{nombre_de_la_normativa_local}}.pp
```

# auditctl

Utilidad para controlar el comportamiento, obtener el estado y gestionar las reglas del Sistema de Auditoría de Linux.

Más información: <https://manned.org/auditctl>.

- Muestra el e[s]tado del sistema de auditoría:

```
sudo auditctl -s
```

- Muestra todas las reglas de auditoría cargadas actualmente:

```
sudo auditctl -l
```

- Elimina todas las reglas de auditoría:

```
sudo auditctl -D
```

- Habilita/deshabilita el sistema de auditoría:

```
sudo auditctl -e {{1|0}}
```

- Vigila un archivo en busca de cambios:

```
sudo auditctl -a always,exit -F arch=b64 -F path={{/ruta/al/archivo}} -F perm=wa
```

- Busca cambios en un directorio de forma recursiva:

```
sudo auditctl -a always,exit -F arch=b64 -F dir={{/ruta/al/directorio/}} -F perm=wa
```

- Muestra ayuda:

```
auditctl -h
```

# aur

Construye paquetes desde el AUR y gestiona repositorios locales.

Nota: Es necesario establecer un repositorio local en **/etc/pacman.conf** e instalar **vim** para que funcione completamente.

Más información: <https://github.com/aurutils/aurutils>.

- Busca un paquete en la base de datos del AUR:

```
aur search {{palabra_clave}}
```

- Descarga un paquete y sus dependencias desde el AUR, los compila y añade a un repositorio local:

```
aur sync {{paquete}}
```

- [l]ista paquetes disponibles en tu repositorio local:

```
aur repo --list
```

- Act[u]aliza los paquetes del repositorio local:

```
aur sync --upgrades
```



# batcat

Este comando es un alias de **bat**.

Más información: <https://github.com/sharkdp/bat>.

- Muestra la documentación del comando original:

```
tldr bat
```

# blkpr

Registra, reserva, libera, anticipa y borra reservas persistentes en un dispositivo de bloque que soporte "Persistent Reservations".

Más información: <https://manned.org/blkpr>.

- Registra (comando) una nueva reserva con una clave dada en un dispositivo determinado:

```
blkpr {{-c|--command}} register {{-k|--key}}  
{{clave_de_reserva}} {{ruta/al/dispositivo}}
```

- Establece el tipo de reserva existente en acceso exclusivo:

```
blkpr -c reserve {{-k|--key}} {{clave_de_reserva}} {{-t|--  
type}} exclusive-access {{ruta/al/dispositivo}}
```

- Adelanta la reserva existente con una clave dada y la reemplaza por una nueva reserva:

```
blkpr -c preempt {{-K|--oldkey}} {{clave_antigua}} {{-k|--  
key}} {{nueva_clave}} {{-t|--type}} write-exclusive {{ruta/  
al/dispositivo}}
```

- Libera una reserva con una clave y [t]ipo dados en un dispositivo determinado:

```
blkpr -c release {{-k|--key}} {{clave_de_reserva}} {{-t|--  
type}} {{tipo_de_reserva}} {{ruta/al/dispositivo}}
```

- Borra todas las reservas de un dispositivo determinado:

```
blkpr -c clear {{-k|--key}} {{clave}} {{ruta/al/dispositivo}}
```

# bspwm

Un gestor de ventanas embaldosado basado en particionado de espacio binario.

Vea también: **bspc**, para controlar el gestor.

Más información: <https://github.com/baskerville/bspwm>.

- Inicia **bspwm** (cabe recalcar que no debe haber otro gestor de ventanas al ejecutar este comando):

```
bspwm -c {{ruta/a/archivo_de_configuración}}
```

# bwa

Herramienta de alineación Burrows-Wheeler.

Mapeador de secuencias de ADN cortas y poco divergentes frente a un gran genoma de referencia, como el genoma humano.

Más información: <https://github.com/lh3/bwa>.

- Indexa el genoma de referencia:

```
bwa index {{ruta/a/referencia.fa}}
```

- Mapea las lecturas de un solo extremo (secuencias) al genoma indexado utilizando 32 subprocesos y comprime el resultado para ahorrar espacio:

```
bwa mem -t 32 {{ruta/a/referencia.fa}} {{ruta/a/lectura_solo_extremo.fq.gz}} | gzip > {{ruta/a/alineamiento_solo_extremo.sam.gz}}
```

- Mapea las lecturas del par final (secuencias) al genoma indexado usando 32 subprocesos y comprime el resultado para ahorrar espacio:

```
bwa mem -t 32 {{ruta/a/referencia.fa}} {{ruta/a/lectura_par_final_1.fq.gz}} {{ruta/a/lectura_par_final_2.fq.gz}} | gzip > {{ruta/a/alineamiento_par_final.sam.gz}}
```

- Mapea las lecturas del par final (secuencias) al genoma indexado usando 32 subprocesos con [M]arcadores divididos más cortos como secundarios para la compatibilidad del archivo SAM de salida con el software Picard y luego comprime el resultado:

```
bwa mem -M -t 32 {{ruta/a/referencia.fa}} {{ruta/a/lectura_par_final_1.fq.gz}} {{ruta/a/lectura_par_final_2.fq.gz}} | gzip > {{ruta/a/alineamiento_par_final.sam.gz}}
```

- Mapea las lecturas finales del par (secuencias) al genoma indexado usando 32 subprocesos con [C]omentarios FASTA/Q (p. ej. BC:Z:CGTAC) anexando a un resultado comprimido:

```
bwa mem -C -t 32 {{ruta/a/referencia.fa}} {{ruta/a/lectura_par_final_1.fq.gz}} {{ruta/a/lectura_par_final_2.fq.gz}} | gzip > {{ruta/a/lectura_par_final.sam.gz}}
```

# cacaclock

Muestra la hora actual como arte ASCII.

Más información: <https://packages.debian.org/sid/caca-utils>.

- Muestra la hora:

```
cacaclock
```

- Cambia la fuente:

```
cacaclock -f {{fuente}}
```

- Cambia el formato usando una especificación de formato de `strftime`:

```
cacaclock -d {{argumentos_strftime}}
```

# cacademo

Muestra una animación aleatoria de arte ASCII.

Más información: <https://packages.debian.org/sid/caca-utils>.

- Ve una animación:

`cacademo`

# cacafire

Muestra un fuego animado ASCII.

Más información: <https://packages.debian.org/sid/caca-utils>.

- Muestra el fuego ASCII:

```
cacafire
```

# cacaview

Muestra una imagen en formato PMN.

Más información: <https://packages.debian.org/sid/caca-utils>.

- Muestra una imagen:

```
cacaview {{ruta/a/imagen}}
```



# cal

Muestra el calendario, con el día actual resaltado.

Más información: <https://manned.org/cal>.

- Muestra el calendario para el mes actual:

```
cal
```

- Muestra el mes anterior, actual y próximo:

```
cal -3
```

- Usa el Lunes como primer día de la semana:

```
cal --monday
```

- Muestra el calendario para un año concreto (4 dígitos):

```
cal {{año}}
```

- Muestra el calendario para un mes y año concretos:

```
cal {{mes}} {{año}}
```

# CC

Este comando es un alias de **gcc**.

Más información: <https://gcc.gnu.org>.

- Muestra la documentación del comando original:

```
tldr gcc
```

# ceph

Un sistema de almacenamiento unificado.

Más información: <https://ceph.io/en>.

- Comprueba el estado del cluster:

```
ceph status
```

- Comprueba las estadísticas de uso del cluster:

```
ceph df
```

- Obtiene las estadísticas de los grupos de colocación de un cluster:

```
ceph pg dump --format {{plain}}
```

- Crea un grupo de almacenamiento:

```
ceph osd pool create {{nombre_pool}} {{número_de_página}}
```

- Elimina un pool de almacenamiento:

```
ceph osd pool delete {{nombre_pool}}
```

- Cambia el nombre de un pool de almacenamiento:

```
ceph osd pool rename {{nombre_actual}} {{nombre_nuevo}}
```

- Auto-repara pool de almacenamiento:

```
ceph pg repair {{nombre_pool}}
```

# check-dfsg-status

Informa de paquetes no libres instalados en sistemas operativos basados en Debian.

Este comando se conocía anteriormente como **vrms**.

Más información: <https://debian.pages.debian.net/check-dfsg-status/>.

- Lista los paquetes no libres y `contrib` (más la descripción):

```
check-dfsg-status
```

- Muestra solo los nombres de los paquetes:

```
check-dfsg-status --sparse
```

# choom

Muestra y cambia el ajuste del OOM-killer.

Más información: <https://manned.org/choom>.

- Muestra la puntuación OOM-killer del proceso con un identificador específico:

```
choom -p {{pid}}
```

- Modifica la puntuación OOM-killer de un proceso específico:

```
choom -p {{pid}} -n {{-1000..+1000}}
```

- Ejecuta un comando con una puntuación OOM-killer específica:

```
choom -n {{-1000..+1000}} {{comando}} {{argumento1 argumento2 ...}}
```

# cmus

Reproductor de música para la línea de comandos.

Usa las teclas de dirección para navegar, **<enter/return>** para seleccionar, y los números 1-8 para cambiar entre las diferentes vistas.

Más información: <https://cmus.github.io>.

- Abre cmus en un directorio concreto:

```
cmus {{ruta/al/directorio}}
```

- Añade un archivo/directorio a la librería:

```
:add {{ruta/al/archivo_o_directorio}}
```

- Pausa/reproduce la canción actual:

```
c
```

- Activa/desactiva modo aleatorio:

```
s
```

- Cierra cmus:

```
q
```

# compseq

Calcula la composición de palabras únicas en secuencias.

Más información: <https://www.bioinformatics.nl/cgi-bin/emboss/help/compseq/>.

- Cuenta frecuencias observadas de palabras en un archivo FASTA, proporcionando valores de parámetros interactivamente:

```
compseq {{ruta/al/archivo.fasta}}
```

- Cuenta frecuencias observadas de pares de aminoácidos en un archivo FASTA, y guarda el resultado en un archivo de texto:

```
compseq {{ruta/al/archivo_proteina.fasta}} -word 2 {{ruta/al/archivo_de_salida.comp}}
```

- Cuenta las frecuencias observadas de hexanucleótidos en un archivo FASTA, luego guarda el resultado en un archivo de texto e ignora los recuentos cero:

```
compseq {{ruta/al/archivo_de_entrada.fasta}} -word 6 {{ruta/al/archivo_de_salida.comp}} -nozero
```

- Cuenta las frecuencias observadas de codones en un marco de lectura concreto, ignorando cualquier recuento superpuesto (es decir, desplaza la ventana en longitud de palabra 3):

```
compseq -sequence {{ruta/al/archivo_de_ingreso_rna.fasta}} -word 3 {{ruta/al/archivo_de_salida.comp}} -nozero -frame {{1}}
```

- Cuenta las frecuencias observadas de codones desplazados en 3 posiciones; ignorando los recuentos superpuestos (debería informar de todos los codones excepto el primero):

```
compseq -sequence {{ruta/al/archivo_de_ingreso_rna.fasta}} -word 3 {{ruta/al/archivo_de_salida.comp}} -nozero -frame 3
```

- Cuenta tripletes de aminoácidos en un archivo FASTA y compara con una ejecución anterior de `compseq` para calcular los valores de frecuencia esperados y normalizados:

```
compseq -sequence {{ruta/al/proteoma_humano.fasta}} -word 3  
{{ruta/al/archivo_salida1.comp}} -nozero -infile {{ruta/al/  
archivo_de_salida2.comp}}
```

- Aproxima el comando anterior sin un archivo previamente preparado, calculando las frecuencias esperadas usando las frecuencias de una sola base/residuo en la(s) secuencia(s) de entrada suministrada(s):

```
compseq -sequence {{ruta/al/proteoma_humano.fasta}} -word 3  
{{ruta/al/archivo_de_salida.comp}} -nozero -calcfreq
```

- Muestra ayuda (utiliza `-help` `-verbose` para obtener más información sobre los calificadores asociados y generales):

```
compseq -help
```



# conky

Monitor de sistema ligero para X.

Más información: <https://github.com/brndnmtthws/conky>.

- Ejecuta con la configuración por defecto:

```
conky
```

- Crea una nueva configuración por defecto:

```
conky -C > ~/.conkyrc
```

- Ejecuta conky con un archivo de configuración concreto:

```
conky -c {{ruta/a/la/configuración}}
```

- Ejecuta en segundo plano (daemon):

```
conky -d
```

- Alinea conky en el escritorio:

```
conky -a {{top|bottom|middle}}_{{left|right|middle}}
```

- Pausa de 5 segundos al iniciar antes de ejecutarlo:

```
conky -p {{5}}
```

# coredumpctl

Recupera y procesa volcados de memoria y sus metadatos.

Más información: <https://www.freedesktop.org/software/systemd/man/coredumpctl.html>.

- Lista todos los volcados de memoria capturados:

```
coredumpctl list
```

- Lista los volcados de memoria capturados para un programa:

```
coredumpctl list {{programa}}
```

- Muestra información sobre los volcados de memoria que coincidan con el PID de un programa:

```
coredumpctl info {{PID}}
```

- Invoca el depurador usando el último volcado de memoria para un programa:

```
coredumpctl debug {{programa}}
```

- Extrae el último volcado de memoria a un fichero:

```
coredumpctl --output={{ruta/al/archivo}} dump {{programa}}
```

# cp

Copia archivos y directorios.

Más información: <https://www.gnu.org/software/coreutils/cp>.

- Copia un archivo a otro directorio:

```
cp {{ruta/al/archivo_origen.ext}} {{ruta/al/
archivo_destino.ext}}
```

- Copia un archivo en otro directorio, conservando el nombre del archivo:

```
cp {{path/to/archivo_origen.ext}} {{ruta/al/
directorio_principal}}
```

- Copia de forma recursiva el contenido de un directorio a otra ubicación (si el destino existe, el directorio es copiado en esa ubicación):

```
cp -r {{ruta/al/directorio_origen}} {{ruta/al/
directorio_destino}}
```

- Copia un directorio de forma recursiva en modo verbose (muestra los archivos a medida que se copian):

```
cp -vr {{ruta/al/directorio_origen}} {{ruta/al/
directorio_destino}}
```

- Copia archivos de texto en otra ubicación en modo interactivo (pregunta al usuario antes de sobrescribir):

```
cp -i {{*.txt}} {{ruta/al/directorio_destino}}
```

- Sigue los enlaces simbólicos antes de copiar:

```
cp -L {{link}} {{ruta/al/directorio_destino}}
```

- Usa la ruta completa de los archivos de origen, creando los directorios intermedios faltantes al copiar:

```
cp --parents {{ruta_de_origen/al/archivo}} {{ruta/al/
archivo_destino}}
```

# cpuid

Muestra información detallada sobre todas las CPUs.

Más información: <https://etallen.com/cpuid.html>.

- Muestra información de todas las CPUs:

```
cpuid
```

- Muestra información sólo para la CPU actual:

```
cpuid -1
```

- Muestra la información hexadecimal en bruto sin decodificar:

```
cpuid -r
```

# cpulimit

Una herramienta para limitar el uso del CPU de otros procesos.

Más información: <https://cpulimit.sourceforge.net/>.

- Limita un proceso existente con PID 1234 para que solo use el 25% del CPU:

```
cpulimit --pid {{1234}} --limit {{25%}}
```

- Limita un programa existente por su nombre de ejecución:

```
cpulimit --exe {{programa}} --limit {{25}}
```

- Ejecuta un programa determinado y limita su uso a solo el 50% del CPU:

```
cpulimit --limit {{50}} -- {{programa argument1 argument2 ...}}
```

- Ejecuta un programa, limita el uso del CPU a 50% y corre cpulimit en segundo plano:

```
cpulimit --limit {{50}} --background -- {{programa}}
```

- Mata su proceso si el uso del CPU del programa supera el 50%:

```
cpulimit --limit 50 --kill -- {{programa}}
```

- Regula su proceso y sus subprocesos para que ninguno supere el 25% del CPU:

```
cpulimit --limit {{25}} --monitor-forks -- {{programa}}
```

# ctop

Visualiza instantáneamente el rendimiento y la salud del contenedor con métricas en tiempo real sobre el uso de CPU, memoria y bloques de IO.

Más información: <https://github.com/bcicen/ctop>.

- Muestra sólo contenedores [a]ctivos:

```
ctop -a
```

- [r]evierte el orden de clasificación de los contenedores:

```
ctop -r
```

- [i]nvierte los colores predeterminados:

```
ctop -i
```

- Muestra ayuda:

```
ctop -h
```

# CU

Llama a otro sistema y actúa como terminal de marcado/serie o realiza transferencias de archivos sin comprobación de errores.

Más información: <https://manned.org/cu>.

- Abre un puerto serie determinado:

```
sudo cu --line {/dev/ttyUSB0}
```

- Abre un puerto serie determinado con una velocidad en baudios determinada:

```
sudo cu --line {/dev/ttyUSB0} --speed {115200}
```

- Abre un puerto serie determinado con una velocidad en baudios determinada y emite un eco de caracteres localmente (modo semidúplex):

```
sudo cu --line {/dev/ttyUSB0} --speed {115200} --halfduplex
```

- Abre un puerto serie determinado con una velocidad en baudios, paridad y sin control de flujo por hardware o software:

```
sudo cu --line {/dev/ttyUSB0} --speed {115200} --parity={par|odd|none} --nortscts --nostop
```

- Sale de la sesión cu cuando está conectado:

```
~.
```

# dd

Convierte y copia un archivo.

Más información: <https://www.gnu.org/software/coreutils/dd>.

- Crea una unidad USB de arranque a partir de un archivo isohybrid (como `archlinux-xxx.iso`) y muestra el progreso:

```
dd if={{ruta/al/archivo.iso}} of={{/dev/unidad_usb}}  
status=progress
```

- Clona una unidad a otra con un tamaño de bloque de 4 MiB y descarga las escrituras antes de que el comando termine:

```
dd bs=4M conv=fsync if={{/dev/unidad_de_origen}} of={{/dev/  
unidad_de_descarga}}
```

- Genera un archivo con un número específico de bytes aleatorios utilizando el controlador aleatorio del kernel:

```
dd bs={{100}} count={{1}} if=/dev/urandom of={{ruta/al/  
archivo_aleatorio}}
```

- Compara el rendimiento de escritura de un disco:

```
dd bs={{1M}} count={{1024}} if=/dev/zero of={{ruta/al/  
fichero_1GB}}
```

- Crea una copia de seguridad del sistema en un archivo IMG (puede restaurarla más tarde intercambiando `if` y `of`), y muestra el progreso:

```
dd if={{/dev/unidad_dispositivo}} of={{ruta/al/archivo.img}}  
status=progress
```

- Comprueba el progreso de una operación `dd` en curso (ejecute este comando desde otro intérprete de comandos):

```
kill -USR1 $(pgrep -x dd)
```



# debuginfod-find

Solicita datos relacionados con debuginfo.

Más información: <https://manned.org/debuginfod-find>.

- Solicita datos basados en `build_id`:

```
debuginfod-find -vv debuginfo {{identificador_de_build}}
```

# dhcpcd

Cliente DHCP.

Más información: <https://roy.marples.name/projects/dhcpcd>.

- Libera todas las direcciones:

```
sudo dhcpcd --release
```

- Solicita nuevas direcciones al servidor DHCP:

```
sudo dhcpcd --rebind
```

# dmesg

Escribe los mensajes del núcleo a la salida estándar.

Más información: <https://manned.org/dmesg>.

- Muestra los mensajes del núcleo:

```
sudo dmesg
```

- Muestra los mensajes de error del núcleo:

```
sudo dmesg --level err
```

- Muestra los mensajes del núcleo y sigue leyendo los nuevos, similar a `tail -f` (disponible en los núcleos 3.5.0 y posteriores):

```
sudo dmesg -w
```

- Muestra cuanta memoria física hay disponible en este sistema:

```
sudo dmesg | grep -i memory
```

- Muestra los mensajes del núcleo, página a página:

```
sudo dmesg | less
```

- Muestra los mensajes del núcleo con una estampilla temporal (disponible en los núcleos 3.5.0 y posteriores):

```
sudo dmesg -T
```

- Muestra los mensajes del núcleo de forma legible para humanos (disponible en los núcleos 3.5.0 y posteriores):

```
sudo dmesg -H
```

- Colorea la salida (disponible en los núcleos 3.5.0 y posteriores):

```
sudo dmesg -L
```

# dmidecode

Muestra la tabla de contenidos del DMI (también conocido como SMBIOS) en un formato legible por humanos.

Requiere privilegios de root.

Más información: <https://manned.org/dmidecode>.

- Muestra todos la tabla de contenidos de DMI:

```
sudo dmidecode
```

- Muestra la versión de la BIOS:

```
sudo dmidecode -s bios-version
```

- Muestra el número de serie del equipo:

```
sudo dmidecode -s system-serial-number
```

- Muestra información de la BIOS:

```
sudo dmidecode -t bios
```

- Muestra información de la CPU:

```
sudo dmidecode -t processor
```

- Muestra información de la memoria:

```
sudo dmidecode -t memory
```

# dnf

Administrador de paquetes para RHEL, CentOS y Fedora (Reemplaza a yum).

Más información: <https://dnf.readthedocs.io>.

- Actualiza todos los paquetes a la última versión disponible:

```
sudo dnf update
```

- Busca un paquete usando palabras clave:

```
dnf search {{palabra_clave}}
```

- Muestra información acerca de un paquete:

```
dnf info {{paquete}}
```

- Instala un nuevo paquete:

```
sudo dnf install {{paquete}}
```

- Instala un nuevo paquete respondiendo sí a todas las preguntas:

```
sudo dnf install -y {{paquete}}
```

- Lista todos los paquetes instalados:

```
dnf list --installed
```

- Encuentra qué paquete provee un archivo determinado:

```
dnf provides {{archivo}}
```

# dnf5

Programa de gestión de paquetes para RHEL, Fedora y CentOS (reemplaza a dnf, que a su vez reemplazó a yum).

DNF5 es una reescritura en C++ del gestor de paquetes DNF con mejor rendimiento y menor tamaño.

Para comandos equivalentes en otros gestores de paquetes, vea <https://wiki.archlinux.org/title/Pacman/Rosetta>.

Más información: <https://dnf5.readthedocs.io>.

- Actualiza los paquetes instalados a las versiones más recientes disponibles:

```
sudo dnf5 upgrade
```

- Busca paquetes mediante palabras clave:

```
dnf5 search {{palabra_clave1 palabra_clave2 ...}}
```

- Muestra detalles sobre un paquete:

```
dnf5 info {{paquete}}
```

- Instala nuevos paquetes (Nota: usa la opción -y para saltar todas las confirmaciones):

```
sudo dnf5 install {{paquete1 paquete2 ...}}
```

- Elimina paquetes:

```
sudo dnf5 remove {{paquete1 paquete2 ...}}
```

- Lista paquetes instalados:

```
dnf5 list --installed
```

- Busca que paquetes proporcionan un comando determinado:

```
dnf5 provides {{comando}}
```

- Elimina o expira los datos almacenados en caché:

```
sudo dnf5 clean all
```

# dos2unix

Cambia saltos de línea con formato DOS a saltos de línea con formato Unix.

Reemplaza CRLF con LF.

Más información: <https://manned.org/dos2unix>.

- Cambia los saltos de línea de un archivo:

```
dos2unix {{nombre_de_archivo}}
```

- Crea una copia con saltos de línea en formato Unix:

```
dos2unix -n {{nombre_de_archivo}} {{nombre_de_archivo}}
```

# dysk

Muestra información del sistema de archivos en una tabla.

Más información: <https://dystroy.org/dysk>.

- Obtén una visión general estándar de tus discos habituales:

```
dysk
```

- Ordena por tamaño libre:

```
dysk --sort free
```

- Incluye solo discos HDD:

```
dysk --filter 'disk = HDD'
```

- Excluye discos SSD:

```
dysk --filter 'disk <> SSD'
```

- Muestra discos con alta ocupación o poco espacio libre:

```
dysk --filter 'use > 65% | free < 50G'
```



# eselect repository

Un módulo **eselect** para configurar repositorios ebuild para Portage.

Después de habilitar un repositorio, tienes que ejecutar **emerge --sync repo\_name** para descargar ebuilds.

Más información: <https://wiki.gentoo.org/wiki/Eselect/Repository>.

- Lista todos los repositorios ebuild registrados en <https://repos.gentoo.org>:

```
eselect repository list
```

- Lista de repositorios habilitados:

```
eselect repository list -i
```

- Habilita un repositorio de la lista según su nombre o índice desde el comando **list**:

```
eselect repository enable {{name|index}}
```

- Activa un repositorio no registrado:

```
eselect repository add {{nombre}} {{rsync|git|mercurial|  
svn|...}} {{sync_uri}}
```

- Deshabilita repositorios sin eliminar su contenido:

```
eselect repository disable {{repo1 repo2 ...}}
```

- Desactiva repositorios y elimina su contenido:

```
eselect repository remove {{repo1 repo2 ...}}
```

- Crea un repositorio local y lo habilita:

```
eselect repository create {{nombre}} {{ruta/al/repo}}
```

# eselect

Herramienta polivalente de configuración y gestión de Gentoo.

Consta de varios módulos que se encargan de tareas administrativas individuales.

Más información: <https://wiki.gentoo.org/wiki/Eselect>.

- Muestra una lista de los módulos instalados:

```
eselect
```

- Vea la documentación de un módulo específico:

```
tldr eselect {{módulo}}
```

- Muestra un mensaje de ayuda para un módulo específico:

```
eselect {{módulo}} help
```

# eu-readelf

Muestra información sobre archivos ELF.

Más información: <https://manned.org/eu-readelf>.

- Muestra toda la información extraíble en un archivo ELF:

```
eu-readelf --all {{ruta/al/archivo}}
```

- Muestra el contenido de todos los segmentos y secciones de NOTE, o de un segmento o sección en particular:

```
eu-readelf --notes[={{.note.ABI-tag}}] {{ruta/al/fichero}}
```

# evtest

Muestra información de los controladores de dispositivos de entrada.

Más información: <https://manned.org/evtest>.

- Lista todos los dispositivos de entrada detectados:

```
sudo evtest
```

- Muestra los eventos de un dispositivo de entrada específico:

```
sudo evtest /dev/input/event{{número}}
```

- Agarra el dispositivo exclusivamente, evitando que otros clientes reciban eventos:

```
sudo evtest --grab /dev/input/event{{número}}
```

- Consulta el estado de una tecla o botón específico en un dispositivo de entrada:

```
sudo evtest --query /dev/input/event{{número}}  
{{tipo_evento}} {{código_evento}}
```

# factorio

Crea e inicia un servidor Factorio sin interfaz gráfica.

Más información: <https://wiki.factorio.com/Multiplayer>.

- Crea un nuevo archivo guardado:

```
{{ruta/a/factorio}} --create {{ruta/a/archivo_guardado.zip}}
```

- Inicia un servidor Factorio:

```
{{ruta/a/factorio}} --start-server {{ruta/a/  
archivo_guardado.zip}}
```

# filefrag

Informa del grado de fragmentación de un archivo en particular.

Más información: <https://manned.org/filefrag>.

- Muestra un informe para uno o más archivos:

```
filefrag {{ruta/al/archivo1 ruta/al/archivo2 ...}}
```

- Muestra un informe con un tamaño de bloque de 1024 bytes:

```
filefrag -k {{ruta/al/archivo}}
```

- Muestra un informe con un tamaño de bloque determinado:

```
filefrag -b{{1024|1K|1M|1G|...}} {{ruta/al/archivo}}
```

- Sincroniza el archivo antes de solicitar la asignación:

```
filefrag -s {{ruta/al/archivo1 ruta/al/archivo2 ...}}
```

- Muestra la asignación de atributos extendidos:

```
filefrag -x {{ruta/al/archivo1 ruta/al/archivo2 ...}}
```

- Muestra un informe con información detallada:

```
filefrag -v {{ruta/al/archivo1 ruta/al/archivo2 ...}}
```

# flatpak run

Ejecuta aplicaciones y tiempos de ejecución flatpak.

Más información: <https://docs.flatpak.org/en/latest/flatpak-command-reference.html#flatpak-run>.

- Ejecuta una aplicación instalada:

```
flatpak run {{com.example.app}}
```

- Ejecuta una aplicación instalada desde una rama específica, por ejemplo, stable, beta, master:

```
flatpak run --branch={{stable|beta|master|...}}  
{{com.example.app}}
```

- Ejecuta un shell interactivo dentro de un flatpak:

```
flatpak run --command={{sh}} {{com.example.app}}
```

# flatpak

Construye, instala y ejecuta aplicaciones y tiempos de ejecución flatpak.

Más información: <https://docs.flatpak.org/en/latest/flatpak-command-reference.html#flatpak>.

- Ejecuta una aplicación instalada:

```
flatpak run {{com.example.app}}
```

- Instala una aplicación desde una fuente remota:

```
flatpak install {{nombre_remoto}} {{com.example.app}}
```

- Lista las aplicaciones instaladas, ignorando los tiempos de ejecución:

```
flatpak list --app
```

- Actualiza todas las aplicaciones y tiempos de ejecución instalados:

```
flatpak update
```

- Añade una fuente remota:

```
flatpak remote-add --if-not-exists {{nombre_remoto}}  
{{url_remota}}
```

- Elimina una aplicación instalada:

```
flatpak remove {{com.example.app}}
```

- Elimina todas las aplicaciones no utilizadas:

```
flatpak remove --unused
```

- Muestra información sobre una aplicación instalada:

```
flatpak info {{com.example.app}}
```



# fpsync

Ejecuta varios procesos de sincronización localmente o en varios trabajadores remotos a través de SSH.

Más información: <https://www.fpart.org/fpsync/>.

- Sincroniza recursivamente un directorio a otra ubicación:

```
fpsync -v {{/ruta/a/origen/}} {{/ruta/a/destino/}}
```

- Sincroniza recursivamente un directorio con la última pasada (activa la opción `--delete` de rsync con cada trabajo de sincronización):

```
fpsync -v -E {{/ruta/a/origen/}} {{/ruta/a/destino/}}
```

- Sincroniza recursivamente un directorio a un destino utilizando 8 trabajos de sincronización simultáneos:

```
fpsync -v -n 8 -E {{/ruta/a/origen/}} {{/ruta/a/destino/}}
```

- Sincroniza recursivamente un directorio a un destino utilizando 8 trabajos de sincronización concurrentes repartidos entre dos trabajadores remotos (máquina1 y máquina2):

```
fpsync -v -n 8 -E -w login@machine1 -w login@machine2 -d {{/ruta/a/directorio/compartido/}} {{/ruta/a/origen/}} {{/ruta/a/destino/}}
```

- Sincroniza recursivamente un directorio a un destino utilizando cuatro trabajadores locales, cada uno transfiriendo como máximo 1000 archivos y 100 MB por trabajo de sincronización:

```
fpsync -v -n 4 -f 1000 -s $((100 * 1024 * 1024)) {{/ruta/a/origen/}} {{/ruta/a/destino/}}
```

- Sincroniza de forma recursiva cualquier directorio pero excluye archivos `.snapshot*` específicos (Nota: las opciones y los valores deben estar separados por un carácter de tubería):

```
fpsync -v -0 "-x|.snapshot*" {{/ruta/a/origen/}} {{/ruta/a/destino/}}
```

# free

Muestra la cantidad de memoria libre y usada en el sistema.

Más información: <https://manned.org/free>.

- Muestra la memoria del sistema:

```
free
```

- Muestra la memoria del sistema en Bytes/KB/MB/GB:

```
free -{{b|k|m|g}}
```

- Muestra la memoria del sistema en unidades legibles por humanos:

```
free -h
```

- Actualiza la salida cada 2 segundos:

```
free -s {{2}}
```

# gedit

Editor de texto del proyecto GNOME.

Más información: <https://help.gnome.org/users/gedit/stable/>.

- Abre un archivo de texto:

```
gedit {{ruta/al/archivo}}
```

- Abre varios archivos de texto:

```
gedit {{archivo1 archivo2 ...}}
```

- Abre un archivo de texto con una codificación específica:

```
gedit --encoding {{UTF-8}} {{ruta/al/archivo}}
```

- Muestra una lista de las codificaciones disponibles:

```
gedit --list-encodings
```

# getenforce

Obtén el modo actual de SELinux (es decir, obligatorio, permisivo o deshabilitado).

Más información: <https://manned.org/getenforce>.

- Muestra el modo actual de SELinux:

```
getenforce
```

# gio trash

Mueve archivos a la papelera.

Usado por GNOME para manejar la papelera.

Más información: <https://manned.org/gio.1>.

- Mueve archivos específicos a la papelera:

```
gio trash {{ruta/al/archivo_o_directorio1 ruta/al/
archivo_o_directorio2 ...}}
```

- Lista los elementos de la papelera:

```
gio trash --list
```

- Restaura un elemento específico de la papelera utilizando su identificador:

```
gio trash trash://{identificador}}
```

# groupdel

Elimina del sistema grupos de usuarios existentes.

Más información: <https://manned.org/groupdel>.

- Borra un grupo existente:

```
sudo groupdel {{nombre_del_grupo}}
```

# grub-mkconfig

Genera un archivo de configuración de GRUB.

Más información: [https://www.gnu.org/software/grub/manual/grub/html\\_node/Invoking-grub\\_002dmkconfig.html](https://www.gnu.org/software/grub/manual/grub/html_node/Invoking-grub_002dmkconfig.html).

- Ejecuta el comando solo e imprime la salida a `stdout`:

```
sudo grub-mkconfig
```

- Genera el archivo de configuración:

```
sudo grub-mkconfig --output={{/boot/grub/grub.cfg}}
```

- Imprime la página de ayuda:

```
grub-mkconfig --help
```

# grubby

Herramienta para configurar los gestores de arranque **grub** y **zipl**.

Más información: <https://manned.org/grubby.8>.

- Añade argumentos de arranque del núcleo a todas las entradas del menú del núcleo:

```
sudo grubby --update-kernel=ALL --args '{{quiet  
console=ttyS0}}'
```

- Elimina los argumentos existentes de la entrada para el núcleo por defecto:

```
sudo grubby --update-kernel=DEFAULT --remove-args {{quiet}}
```

- Lista todas las entradas del menú del núcleo:

```
sudo grubby --info=ALL
```



# halt

Detiene, apaga o reinicia la máquina.

Más información: <https://manned.org/halt.8>.

- Detiene la máquina:

```
halt
```

- Apaga la máquina:

```
halt --poweroff
```

- Reinicia la máquina:

```
halt --reboot
```

# hyprctl

Controla partes del compositor Hyprland Wayland.

Más información: <https://wiki.hyprland.org/Configuring/Using-hyprctl>.

- Recarga la configuración de Hyprland:

```
hyprctl reload
```

- Muestra el nombre de la ventana activa:

```
hyprctl activewindow
```

- Lista todos los dispositivos de entrada conectados:

```
hyprctl devices
```

- Lista todas las salidas con sus respectivas propiedades:

```
hyprctl workspaces
```

- Llama a un gestor con un argumento:

```
hyprctl dispatch exec {{aplicación}}
```

- Establece una palabra clave de configuración de forma dinámica:

```
hyprctl keyword {{palabra_clave}} {{valor}}
```

- Muestra versión:

```
hyprctl version
```

# hyprpm

Complementos de control para el compositor Hyprland Wayland.

Más información: <https://wiki.hyprland.org/Plugins/Using-Plugins/#hyprpm>.

- Añade un complemento:

```
hyprpm add {{url_de_repositorio_git}}
```

- Elimina un complemento:

```
hyprpm remove {{url_de_repositorio_git|  
nombre_del_complemento}}
```

- Activa un complemento:

```
hyprpm enable {{nombre_del_complemento}}
```

- Desactiva un complemento:

```
hyprpm disable {{nombre_del_complemento}}
```

- Actualiza y comprueba todos los complementos:

```
hyprpm update
```

- Fuerza una operación:

```
hyprpm {{-f|--force}} {{operación}}
```

- Lista todos los complementos instalados:

```
hyprpm list
```

# i3

Un gestor dinámico de ventanas en mosaico.

Más información: <https://i3wm.org/docs/userguide.html>.

- Comienza i3 (Tener en cuenta que no debe haber abierto ningún otro gestor de ventanas existente cuando se ejecute este comando):

```
i3
```

- Abre una terminal en una nueva ventana:

```
<Super> + <Enter>
```

- Crea un nuevo espacio de trabajo:

```
<Super> + <Shift> + {{número}}
```

- Cambia al espacio de trabajo número n:

```
<Super> + {{n}}
```

- Abre una nueva ventana en mosaico horizontal:

```
<Super> + h
```

- Abre una nueva ventana en mosaico vertical:

```
<Super> + v
```

- Abre una aplicación (escribir el nombre de la aplicación después de ejecutar el comando):

```
<Super> + D
```

# i7z

Una herramienta de informes en tiempo real para CPUs Intel (sólo i3, i5 e i7).

Más información: <https://manned.org/i7z>.

- Inicia i7z (se necesita ejecutar en modo superusuario):

```
sudo i7z
```

# ifdown

Desactiva interfaces de red.

Más información: <https://manned.org/ifdown>.

- Desactiva la interfaz eth0:

```
ifdown {{eth0}}
```

- Desactiva todas las interfaces que estén activadas:

```
ifdown -a
```

# ip route list

Este comando es un alias de **ip route show**.

- Muestra la documentación del comando original:

```
tldr ip-route-show
```

# iwlist

Obtén información detallada de una interfaz inalámbrica.

Más información: <https://manned.org/iwlist.8>.

- Muestra la lista de puntos de acceso y células ad-hoc en alcance:

```
iwlist {{interfaz_inalámbrica}} scan
```

- Muestra las frecuencias disponibles en el dispositivo:

```
iwlist {{interfaz_inalámbrica}} frequency
```

- Lista las tasas de bits soportadas por el dispositivo:

```
iwlist {{interfaz_inalámbrica}} rate
```

- Enumera los parámetros de autenticación WPA configurados actualmente:

```
iwlist {{interfaz_inalámbrica}} auth
```

- Enumera todas las claves de cifrado WPA configuradas en el dispositivo:

```
iwlist {{interfaz_inalámbrica}} wpakeys
```

- Enumera los tamaños de clave de cifrado admitidos y todas las claves de cifrado configuradas en el dispositivo:

```
iwlist {{interfaz_inalámbrica}} keys
```

- Enumera los distintos atributos y modos de gestión de energía del dispositivo:

```
iwlist {{interfaz_inalámbrica}} power
```

- Enumera los elementos de información genéricos configurados en el dispositivo (utilizados para la compatibilidad con WPA):

```
iwlist {{interfaz_inalámbrica}} genie
```



# journalctl

Consulta el registro systemd.

Más información: <https://manned.org/journalctl>.

- Muestra todos los mensajes con nivel de prioridad 3 (errores) de este [b]oot:

```
journalctl -b --priority=3
```

- Elimina los registros diarios con más de 2 días de antigüedad:

```
journalctl --vacuum-time=2d
```

- Muestra solo las últimas N líneas y sigue los mensajes nuevos (como `tail -f` de un syslog tradicional):

```
journalctl --lines {{N}} --follow
```

- Muestra todos los mensajes de una [u]nidad específica:

```
journalctl --unit {{unidad}}
```

- Muestra los registros de una unidad determinada desde la última vez que se inició:

```
journalctl _SYSTEMD_INVOCATION_ID=$(systemctl show --value --property=InvocationID {{unidad}})
```

- Filtra mensajes dentro de un intervalo de tiempo (marca de tiempo o marcadores de posición como "ayer"):

```
journalctl --since {{now|today|yesterday|tomorrow}} --until "{{YYYY-MM-DD HH:MM:SS}}"
```

- Muestra todos los mensajes de un proceso específico:

```
journalctl _PID={{pid}}
```

- Mostrar todos los mensajes de un ejecutable específico:

```
journalctl {{ruta/al/ejecutable}}
```

# kde-builder

Construye fácilmente componentes de KDE desde tus repositorios fuente.

Sustituye a **kdesrc-build**.

Más información: <https://kde-builder.kde.org/en/cmdline/cmdline-usage.html>.

- Inicializa kde-builder:

```
kde-builder --initial-setup
```

- Compila un componente KDE y sus dependencias desde el código fuente:

```
kde-builder {{nombre_componente}}
```

- Compila un componente sin actualizar el código local y sin compilar sus dependencias:

```
kde-builder --no-src --no-include-dependencies  
{{nombre_componente}}
```

- Actualiza los directorios de compilación antes de compilar:

```
kde-builder --refresh-build {{nombre_componente}}
```

- Reanuda la compilación a partir de una dependencia determinada:

```
kde-builder --resume-from={{componente_de_dependencia}}  
{{nombre_componente}}
```

- Ejecuta un componente con un nombre de ejecutable determinado:

```
kde-builder --run {{nombre_ejecutable}}
```

- Construye todos los componentes configurados:

```
kde-builder
```

- Utiliza las bibliotecas del sistema en lugar de un componente si no se puede compilar:

```
kde-builder --no-stop-on-failure {{nombre_componente}}
```

# last

Lista la información de los últimos inicios de sesión de usuario.

Vea también: **lastb**, **login**.

Más información: <https://manned.org/last.1>.

- Lista la información de inicio de sesión (por ejemplo, nombre de usuario, terminal, tiempo de arranque, núcleo) de todos los usuarios:

```
last
```

- Lista la información de inicio de sesión de un usuario específico:

```
last {{nombre_de_usuario}}
```

- Muestra la información de una terminal específica:

```
last {{tty1}}
```

- Lista la información actualizada (por defecto, la más reciente aparece al principio):

```
last | tac
```

- Muestra información sobre el arranque del sistema:

```
last "{{system boot}}"
```

- Lista información con un formato de [t]iempo específico:

```
last --time-format {{notime|full|iso}}
```

- Lista información desde una fecha y hora específica:

```
last --since {{-7days}}
```

- Muestra información (es decir, nombre e IP) de hosts remotos:

```
last --dns
```

# libuser-lid

Muestra los grupos de un usuario o los usuarios de un grupo.

En Fedora y Arch Linux, este programa se instala como **lid**.

Más información: <https://manned.org/lid.8>.

- Lista los grupos primarios y secundarios de un usuario específico:

```
sudo lid {{nombre_de_usuario}}
```

- Lista los usuarios de un grupo específico:

```
sudo lid --group {{nombre}}
```

# lid

NOTA: Esta página es actualmente una de redirección. Si está familiarizado con este programa, por favor abra una solicitud de extracción.

Consulta la base de datos de identificadores y reporta los resultados.

En Fedora y Arch Linux, **lid** es otro programa. Vea **tldr libuser-lid**.

Más información: <https://www.gnu.org/software/ident/>.

- Ve documentación de `libuser-lid`:

```
tldr libuser-lid
```

# line

Lee una única línea de entrada.

Más información: <https://manned.org/line.1>.

- Lee una entrada:

```
line
```

# lsb\_release

Proporciona información específica de la distribución y LSB (Linux Standard Base).

Más información: [https://manned.org/lsb\\_release](https://manned.org/lsb_release).

- Muestra toda la información disponible:

```
lsb_release -a
```

- Muestra una descripción del sistema operativo (normalmente el nombre completo):

```
lsb_release -d
```

- Muestra solo el nombre del sistema operativo (ID) sin el campo nombre:

```
lsb_release -i -s
```

- Muestra el número de versión y el nombre en clave de la distribución sin el campo de nombre:

```
lsb_release -rcs
```

# lsmod

Muestra el estado de los módulos cargados en el kernel de linux.

Vease tambien **modprobe**, el cual carga módulos de kernel.

Más información: <https://manned.org/lsmod>.

- Lista todos los módulos de kernel cargados:

```
lsmod
```



# lsusb

Muestra información sobre los buses USB y los dispositivos conectados a ellos.

Más información: <https://manned.org/lsusb>.

- Muestra todos los dispositivos USB disponibles:

```
lsusb
```

- Lista la jerarquía USB como un árbol:

```
lsusb -t
```

- Muestra información detallada sobre los dispositivos USB:

```
lsusb --verbose
```

- Muestra información detallada sobre un dispositivo USB:

```
lsusb --verbose -s {{bus}}:{{número de dispositivo}}
```

- Muestra sólo los dispositivos con un ID de proveedor y producto determinados:

```
lsusb -d {{vendedor}}:{{producto}}
```

# man

Da formato y muestra páginas del manual.

Más información: <https://manned.org/man>.

- Muestra la página del manual para un comando:

```
man {{comando}}
```

- Muestra la página del manual para un comando de la sección 7:

```
man {{7}} {{comando}}
```

- Lista todas las secciones disponibles para un comando:

```
man --what-is {{comando}}
```

- Muestra las rutas usadas para la búsqueda de las páginas:

```
man --path
```

- Muestra la ubicación de la página del manual en lugar de la propia página:

```
man --where {{comando}}
```

- Muestra la página del manual usando una ubicación específica:

```
man --locale={{locale}} {{comando}}
```

- Busca las páginas del manual que contienen la cadena indicada:

```
man --apropos "{{cadena_a_buscar}}"
```

# megadl

Este comando es un alias de **megatools-dl**.

Más información: <https://megatools.megous.com/man/megatools-dl.html>.

- Muestra la documentación del comando original:

```
tldr megatools-dl
```

# mkfs.erofs

Crea un sistema de archivos EROFS en una imagen.

Más información: <https://erofs.docs.kernel.org/en/latest/>.

- Crea un sistema de archivos EROFS basado en el directorio raíz:

```
mkfs.erofs image.erofs root/
```

- Crea una imagen EROFS con un UUID específico:

```
mkfs.erofs -U {{UUID}} image.erofs root/
```

- Crea una imagen EROFS comprimida:

```
mkfs.erofs -z lz4hc image.erofs root/
```

- Crea una imagen EROFS en la que todos los archivos pertenezcan a root:

```
mkfs.erofs --all-root image.erofs root/
```

# mkfs.f2fs

Crea un sistema de archivos F2FS en una partición.

Más información: <https://manned.org/mkfs.f2fs>.

- Crea un sistema de archivos F2FS en la primera partición del dispositivo b (sdb1):

```
sudo mkfs.f2fs {{/dev/sdb1}}
```

- Crea un sistema de archivos F2FS con una etiqueta de volumen:

```
sudo mkfs.f2fs -l {{etiqueta_volumen}} {{/dev/sdb1}}
```

# mklost+found

Crea un directorio lost+found.

Más información: <https://manned.org/mklost+found>.

- Crea un directorio `lost+found` en el directorio actual:

```
mklost+found
```

# mopac

MOPAC (Molecular Orbital PACkage) es un programa semiempírico de química cuántica basado en la aproximación NDDO de Dewar y Thiel.

Más información: <https://github.com/openmopac/mopac>.

- Realiza los cálculos a partir de un archivo de entrada (.mop, .dat y .arc):

```
mopac {{ruta/al/archivo_de_entrada}}
```

- Mínimo ejemplo de trabajo con HF que escribe en el directorio actual y lo guarda en el archivo de salida:

```
touch test.out; echo "PM7\n#comment\n\nH 0.95506 0.05781  
-0.03133\nF 1.89426 0.05781 -0.03133" > test.mop; mopac  
test.mop & tail -f test.out
```

# mount.ddi

Monta imágenes de disco reconocibles.

Vea **tldr systemd-dissect** para otros comandos relevantes para DDIs.

Más información: <https://www.freedesktop.org/software/systemd/man/latest/systemd-dissect.html>.

- Monta una imagen con un sistema operativo:

```
mount.ddi {{ruta/a/imagen.raw}} {{/mnt/image}}
```



# nautilus

Explorador de archivos por defecto para el entorno de escritorio GNOME.

También conocido como Archivos de GNOME.

Más información: <https://manned.org/nautilus>.

- Inicia Nautilus:

```
nautilus
```

- Inicia Nautilus como usuario root:

```
nautilus admin:/
```

- Inicia Nautilus y muestra un directorio específico:

```
nautilus {{ruta/al/directorio}}
```

- Inicia Nautilus con un archivo o directorio específico seleccionado:

```
nautilus --select {{ruta/al/archivo_o_directorio}}
```

- Inicia Nautilus en una ventana separada:

```
nautilus --new-window
```

- Cierra todas las instancias de Nautilus:

```
nautilus --quit
```

- Muestra la ayuda:

```
nautilus --help
```

# ncal

Este comando es un alias de **cal**.

Más información: <https://manned.org/ncal>.

- Muestra la documentación del comando original:

```
tldr cal
```

# newgrp

Cambia el grupo primario de pertenencia.

Más información: <https://manned.org/newgrp>.

- Cambia el grupo primario de pertenencia del usuario:

```
newgrp {{nombre_grupo}}
```

- Restablece el grupo primario de pertenencia al grupo por defecto del usuario  
/etc/passwd:

```
newgrp
```

# nsenter

Ejecuta un nuevo comando en el espacio de nombres de un proceso en ejecución.

Particularmente útil para imágenes Docker o jaulas chroot.

Más información: <https://manned.org/nsenter>.

- Ejecuta un comando específico utilizando los mismos espacios de nombres como un proceso existente:

```
nsenter --target {{pid}} --all {{comando}}  
{{argumentos_del_comando}}
```

- Ejecuta un comando específico en el espacio de nombres mount|UTS|IPC|network|PID|user|cgroup|time de un proceso existente:

```
nsenter --target {{pid}} --{{mount|uts|ipc|net|pid|user|  
cgroup}} {{comando}} {{argumentos_del_comando}}
```

- Ejecuta un comando específico en los espacios de nombres UTS, time e IPC de un proceso existente:

```
nsenter --target {{pid}} --uts --time --ipc -- {{comando}}  
{{argumentos_del_comando}}
```

- Ejecuta un comando específico en el espacio de nombres de un proceso existente haciendo referencia a procfs:

```
nsenter --pid=/proc/{{pid}}/pid/net -- {{comando}}  
{{argumentos_del_comando}}
```

# ntfsfix

Arregla problemas habituales de una partición NTFS.

Más información: <https://manned.org/ntfsfix>.

- Arregla una partición NTFS dada:

```
sudo ntfsfix {{/dev/sdXN}}
```

# ntpd

El daemon oficial de NTP (Network Time Protocol) para sincronizar el reloj del sistema con servidores de tiempo remotos o relojes de referencia locales.

Más información: <https://manned.org/ntpd>.

- Inicia el daemon:

```
sudo ntpd
```

- Sincroniza la hora del sistema con servidores remotos una sola vez (sale después de sincronizar):

```
sudo ntpd --quit
```

- Sincroniza una sola vez permitiendo "grandes" ajustes:

```
sudo ntpd --panicgate --quit
```

# objcopy

Copia el contenido de un archivo de objetos a otro archivo.

Más información: <https://manned.org/objcopy>.

- Copia datos a otro archivo:

```
objcopy {{ruta/al/archivo_de_origen}} {{ruta/al/
archivo_de_destino}}
```

- Traduce ficheros de un formato a otro:

```
objcopy --input-target={{formato_de_entrada}} --output-target
{{formato_de_salida}} {{ruta/al/archivo_de_origen}} {{ruta/
al/archivo_de_destino}}
```

- Elimina toda la información de símbolos del archivo:

```
objcopy --strip-all {{ruta/al/archivo_de_origen}} {{ruta/al/
archivo_de_destino}}
```

- Elimina la información de depuración del archivo:

```
objcopy --strip-debug {{ruta/al/archivo_de_origen}} {{ruta/
al/archivo_de_destino}}
```

- Copia una sección específica del archivo de origen al archivo de destino:

```
objcopy --only-section {{section}} {{ruta/al/
archivo_de_origen}} {{ruta/al/archivo_de_destino}}
```

# pacgraph

Dibuja un gráfico de los paquetes instalados a PNG/SVG/GUI/consola.

Más información: <https://github.com/keenerd/pacgraph>.

- Crea un gráfico SVG y PNG:

```
pacgraph
```

- Crear un gráfico SVG:

```
pacgraph --svg
```

- Imprime resumen en la consola:

```
pacgraph --console
```

- Anula el nombre de archivo/ubicación por defecto (Nota: No especifiques la extensión del archivo):

```
pacgraph --file={{ruta/al/archivo}}
```

- Cambia el color de los paquetes que no son dependencias:

```
pacgraph --top={{color}}
```

- Cambia el color de los paquetes dependientes:

```
pacgraph --dep={{color}}
```

- Cambia el color de fondo de un gráfico:

```
pacgraph --background={{color}}
```

- Cambia el color de los enlaces entre paquetes:

```
pacgraph --link={{color}}
```



# pacman --database

Opera en la base de datos de paquetes de Arch Linux.

Modifica ciertos atributos de los paquetes instalados.

Vea también: **pacman**.

Más información: <https://manned.org/pacman.8>.

- Marca un paquete como instalado implícitamente:

```
sudo pacman --database --asdeps {{paquete}}
```

- Marca un paquete como instalado explícitamente:

```
sudo pacman --database --asexplicit {{paquete}}
```

- Verifica que todas las dependencias del paquete estén instaladas:

```
pacman --database --check
```

- Verifica los repositorios para asegurarse de que todas las dependencias especificadas estén disponibles:

```
pacman --database --check --check
```

- Muestra solo mensajes de error:

```
pacman --database --check --quiet
```

- Muestra ayuda:

```
pacman --database --help
```

# pkgadd

Añade un paquete a un sistema CRUX.

Más información: [https://docs.oracle.com/cd/E88353\\_01/html/E72487/pkgadd-8.html](https://docs.oracle.com/cd/E88353_01/html/E72487/pkgadd-8.html).

- Instala un paquete de software local:

```
pkgadd {{nombre_paquete}}
```

- Actualiza un paquete ya instalado a partir de un paquete local:

```
pkgadd -u {{nombre_paquete}}
```

# pkgrm

Elimina un paquete de un sistema CRUX.

Más información: [https://docs.oracle.com/cd/E88353\\_01/html/E72487/pkgrm-8.html](https://docs.oracle.com/cd/E88353_01/html/E72487/pkgrm-8.html).

- Elimina un paquete instalado:

```
pkgrm {{nombre_del_paquete}}
```

# plasmashell

Inicia y reinicia Plasma Desktop.

Más información: <https://invent.kde.org/plasma/plasma-desktop>.

- Reinicia plasmashell:

```
systemctl restart --user plasma-plasmashell
```

- Reinicia plasmashell sin systemd:

```
plasmashell --replace & disown
```

- Muestra ayuda en las opciones de la línea de comandos:

```
plasmashell --help
```

- Muestra la ayuda, incluidas las opciones de Qt:

```
plasmashell --help-all
```

# portageq

Consulta información sobre Portage, el gestor de paquetes de Gentoo Linux.

Las variables de entorno específicas de Portage que se pueden consultar están listadas en **/var/db/repos/gentoo/profiles/info\_vars**.

Más información: <https://wiki.gentoo.org/wiki/Portageq>.

- Muestra el valor de una variable de entorno específica de Portage:

```
portageq envvar {{variable}}
```

- Muestra una lista detallada de los repositorios configurados con Portage:

```
portageq repos_config /
```

- Muestra una lista de repositorios ordenados por prioridad (el más alto, primero):

```
portageq get_repos /
```

- Muestra un fragmento específico de metadatos sobre un átomo (por ejemplo, el nombre del paquete incluyendo la versión):

```
portageq metadata / {{ebuild|porttree|binary|...}}  
{{categoría}}/{{paquete}} {{BDEPEND|DEFINED_PHASES|  
DEPEND|...}}
```

# poweroff

Apaga el sistema.

Más información: <https://www.manned.org/poweroff>.

- Apaga el sistema:

```
poweroff
```

- Detén el sistema (igual que `halt`):

```
poweroff --halt
```

- Reinicia el sistema (igual que `reboot`):

```
poweroff --reboot
```

- Apaga inmediatamente el sistema sin contactar al administrador:

```
poweroff --force
```

- Escribe una entrada en el archivo `wtmp` sin apagar el sistema:

```
poweroff --wtmp-only
```

# prime-run

Ejecuta un programa utilizando una tarjeta gráfica Nvidia alternativa.

Más información: [https://wiki.archlinux.org/title/PRIME#PRIME\\_render\\_offload](https://wiki.archlinux.org/title/PRIME#PRIME_render_offload).

- Ejecuta un programa utilizando una GPU Nvidia dedicada:

```
prime-run {{comando}}
```

- Valida si se está utilizando la tarjeta Nvidia:

```
prime-run glxinfo | grep "OpenGL renderer"
```

# proctl

Gestiona licencias de proyectos e idiomas, cambiar entre licencias de plantillas.

Más información: <https://github.com/HeCodes2Much/proctl>.

- Lista licencias disponibles:

```
proctl {{-ll|-list-licenses}}
```

- Lista de idiomas disponibles:

```
proctl {{-lL|-lista-idiomos}}
```

- Selecciona una licencia en un menú FZF:

```
proctl {{-pl|-elegir-licencia}}
```

- Selecciona un idioma en un menú FZF:

```
proctl {{-pL|-elegir-idioma}}
```

- Elimina todas las licencias del proyecto actual:

```
proctl {{-r|-quitar-licencia}}
```

- Crea una nueva plantilla de licencia:

```
proctl {{-t|-nueva-plantilla}}
```

- Elimina una licencia de las plantillas:

```
proctl {{-R|-elimina-licencia}} {{@nombre_licencia1  
@nombre_licencia2 ...}}
```

- Muestra esta lista de comandos:

```
proctl {{-h|-ayuda}}
```



# pw-config

Enumera las rutas y secciones de configuración que utilizarán el servidor y los clientes de PipeWire.

Más información: [https://docs.pipewire.org/page\\_man\\_pw-config\\_1.html](https://docs.pipewire.org/page_man_pw-config_1.html).

- Lista todos los archivos de configuración que se utilizarán:

```
pw-config
```

- Lista todos los archivos de configuración que utilizará el servidor PulseAudio de PipeWire:

```
pw-config --name pipewire-pulse.conf
```

- Lista todas las secciones de configuración utilizadas por el servidor PulseAudio de PipeWire:

```
pw-config --name pipewire-pulse.conf list
```

- Lista los fragmentos `context.properties` utilizados por los clientes JACK:

```
pw-config --name jack.conf list context.properties
```

- Lista las `context.properties` fusionadas utilizadas por los clientes JACK:

```
pw-config --name jack.conf merge context.properties
```

- Lista los `context.modules` fusionados utilizados por el servidor PipeWire y `[r]eformat`:

```
pw-config --name pipewire.conf --recurse merge  
context.modules
```

- Muestra la ayuda:

```
pw-config --help
```

# pw-metadata

Supervisa, establece y elimina metadatos en objetos PipeWire.

Vea también: **pipewire**, **pw-mon**, **pw-cli**.

Más información: [https://docs.pipewire.org/page\\_man\\_pw-metadata\\_1.html](https://docs.pipewire.org/page_man_pw-metadata_1.html).

- Muestra metadatos en el formato por defecto:

```
pw-metadata
```

- Muestra metadatos con el identificador 0 en `settings`:

```
pw-metadata {{-n|--name}} {{settings}} {{0}}
```

- Lista todos los objetos de metadatos disponibles:

```
pw-metadata {{-l|--list}}
```

- Continúa ejecutando y registrando los cambios en los metadatos:

```
pw-metadata {{-m|--monitor}}
```

- Elimina todos los metadatos:

```
pw-metadata {{-d|--delete}}
```

- Ajusta `log.level` a 1 en `settings`:

```
pw-metadata --name {{settings}} {{0}} {{log.level}} {{1}}
```

- Muestra ayuda:

```
pw-metadata --help
```

# pwdx

Muestra el directorio de trabajo de un proceso.

Más información: <https://manned.org/pwdx>.

- Muestra el directorio de trabajo actual de un proceso:

```
pwdx {{identificador_de_proceso}}
```

# pwn

Biblioteca de desarrollo de exploits diseñada para la creación rápida de prototipos.

Más información: <https://docs.pwntools.com/en/stable/commandline.html>.

- Convierte código ensamblador a bytes:

```
pwn asm "{{xor edi, edi}}"
```

- Crea un patrón cíclico con un número específico de caracteres:

```
pwn cyclic {{número}}
```

- Codifica datos en el sistema hexadecimal:

```
pwn hex {{deafbeef}}
```

- Decodifica datos en hexadecimal:

```
pwn unhex {{6c4f7645}}
```

- Imprime código de intérprete de Linux x64 para ejecutar en un intérprete:

```
pwn shellcraft {{amd64.linux.sh}}
```

- Comprueba la configuración de seguridad binaria del archivo ELF dado:

```
pwn checksec {{ruta/al/archivo}}
```

- Busca actualizaciones de Pwntools:

```
pwn update
```

- Muestra la versión:

```
pwn version
```

# raspinfo

Muestra información del sistema en una Raspberry Pi.

Más información: <https://github.com/raspberrypi/Utils/tree/master/raspinfo>.

- Muestra información del sistema:

```
raspinfo
```

# readelf

Muestra información sobre archivos ELF.

Más información: <https://manned.org/readelf.1>.

- Muestra toda la información de un archivo ELF:

```
readelf -all {{ruta/al/archivo_binario}}
```

- Muestra todas las cabeceras presentes en un archivo ELF:

```
readelf --headers {{ruta/al/archivo_binario}}
```

- Muestra las entradas en la sección de la tabla de símbolos del archivo ELF, si tiene una:

```
readelf --symbols {{ruta/al/archivo_binario}}
```

- Muestra la información de la cabecera ELF:

```
readelf --file-header {{ruta/al/archivo_binario}}
```

- Muestra información de cabecera de la sección ELF:

```
readelf --section-headers {{ruta/al/archivo_binario}}
```

# reboot

Reinicia la máquina.

Más información: <https://manned.org/reboot.8>.

- Reinicia inmediatamente:

```
reboot
```

- Reinicia inmediatamente sin hacer un apagado limpio:

```
reboot -f
```

# repo-remove

Utilidad de mantenimiento de la base de datos de paquetes que elimina paquetes de un repositorio local.

Más información: <https://manned.org/repo-add>.

- Elimina un paquete de un repositorio local:

```
repo-remove {{ruta/a/base_de_datos.db.tar.gz}} {{paquete}}
```



# reset

Reinicializa la terminal actual. Borra toda la pantalla de la terminal.

Más información: <https://manned.org/reset>.

- Reinicializa la terminal actual:

```
reset
```

- Muestra el tipo de terminal:

```
reset -q
```

# rig

Utilidad para generar un nombre, apellido, calle y número, junto a ubicación geográfica consistente (un conjunto válido de ciudad, estado y código postal).

Más información: <https://manned.org/rig>.

- Muestra un nombre aleatorio (masculino o femenino) y una dirección:

```
rig
```

- Muestra un nombre [m]asculino, (o [f]emenino) aleatorio y una dirección:

```
rig -{{m|f}}
```

- Usa archivos de datos de un directorio específico (por defecto es `/usr/share/rig`):

```
rig -d {{ruta/al/directorio}}
```

- Especifica el número de identidades a generar:

```
rig -c {{numero}}
```

- Especifica el número de identidades femininas a generar:

```
rig -f -c {{numero}}
```

# rpi-otp-private-key

Muestra la llave privada de la clave de un solo uso (OTP) de una Raspberry Pi.

Más información: <https://www.raspberrypi.com/documentation/computers/raspberry-pi.html#program-a-key-into-otp-with-rpi-otp-private-key>.

- Lee la clave privada OTP:

```
rpi-otp-private-key
```

# rpmconf

Gestiona los archivos RPMNEW, RPMSAVE y RPMORIG creados por las actualizaciones de paquetes.

Vea también: **rpm**.

Más información: <https://github.com/xsuchy/rpmconf>.

- Lista los archivos sobrantes y elige interactivamente que hacer con cada uno de ellos:

```
sudo rpmconf --all
```

- Elimina los archivos huérfanos RPMNEW y RPMSAVE:

```
sudo rpmconf --all --clean
```

# run0

Eleva privilegios interactivamente.

Similar a **sudo**, pero no es un binario SUID, la autenticación tiene lugar a través de polkit, y los comandos se invocan desde un servicio **systemd**.

Más información: <https://www.freedesktop.org/software/systemd/man/latest/run0.html>.

- Ejecuta un comando como root:

```
run0 {{comando}}
```

- Ejecuta un comando como otro usuario y/o grupo:

```
run0 {{-u|--user}} {{nombre_de_usuario|uid}} {{-g|--group}}  
{{nombre_de_grupo|gid}} {{comando}}
```

# sbctl

Un gestor de claves de arranque seguro fácil de usar.

Nota: no registrar los certificados de Microsoft puede bloquear su sistema. Vea <https://github.com/Foxboron/sbctl/wiki/FAQ#option-rom>.

Más información: <https://github.com/Foxboron/sbctl#usage>.

- Muestra el estado actual del arranque seguro:

```
sbctl status
```

- Crea claves de arranque seguro personalizadas (todo se almacena en `/var/lib/sbctl`):

```
sbctl create-keys
```

- Inscribe las claves de arranque seguro personalizadas y los certificados de proveedor UEFI de Microsoft:

```
sbctl enroll-keys --microsoft
```

- Firma un binario EFI con la clave creada y guarda el archivo en la base de datos:

```
sbctl sign {{-s|--save}} {{ruta/al/binario_efi}}
```

- Vuelve a firmar todos los archivos guardados:

```
sbctl sign-all
```

- Comprueba que se han firmado todos los ejecutables EFI de la partición EFI del sistema:

```
sbctl verify
```

# sensible-browser

Abre el navegador predeterminado.

Más información: <https://manned.org/sensible-browser>.

- Abre una nueva ventana del navegador predeterminado:

```
sensible-browser
```

- Abre una URL en el navegador predeterminado:

```
sensible-browser {{url}}
```

# sensors

Proporciona información de los sensores.

Más información: <https://manned.org/sensors>.

- Muestra las lecturas actuales de todos los sensores:

```
sensors
```

- Muestra las temperaturas en grados Fahrenheit:

```
sensors --fahrenheit
```



# service

Gestiona los servicios mediante la ejecución de scripts init.

Se debe omitir la ruta completa del script (se asume `/etc/init.d/`).

Más información: <https://manned.org/service>.

- Lista el nombre y el estado de todos los servicios:

```
service --status-all
```

- Inicia/Para/Reinicia/Recarga servicio (start/stop debería estar siempre disponible):

```
service {{nombre_de_servicio}} {{start|stop|restart|reload}}
```

- Hace un reinicio completo (ejecuta el script dos veces con start y stop):

```
service {{nombre_de_servicio}} --full-restart
```

- Muestra el estado actual de un servicio:

```
service {{nombre_de_servicio}} status
```

# showkey

Muestra el código de las teclas pulsadas en el teclado, útil para depurar problemas relacionados con el teclado y la reasignación de teclas.

Más información: <https://manned.org/showkey>.

- Visualiza códigos de teclas en decimal:

```
sudo showkey
```

- Visualiza códigos de rastreo en hexadecimal:

```
sudo showkey {{-s|--scancodes}}
```

- Muestra códigos de teclas en decimal (por defecto):

```
sudo showkey {{-k|--keycodes}}
```

- Muestra los códigos en ASCII, decimal y hexadecimal:

```
sudo showkey {{-a|--ascii}}
```

- Sal del programa:

```
Ctrl + d
```

# shutdown

Apaga y reinicia el sistema.

Más información: <https://manned.org/shutdown.8>.

- Apaga ([h]alt) inmediatamente:

```
shutdown -h now
```

- [r]einicia inmediatamente:

```
shutdown -r now
```

- [r]einicia en 5 minutos:

```
shutdown -r +{{5}} &
```

- Apaga a las 01:00 pm (Usa el reloj de 24[h]):

```
shutdown -h 13:00
```

- [c]ancela una operación pendiente de apagado/reinicio:

```
shutdown -c
```

# slurp

Selecciona una región en un compositor Wayland.

Más información: <https://github.com/emersion/slurp>.

- Selecciona una región y la imprime en `stdout`:

```
slurp
```

- Selecciona una región e imprímela en `stdout`, mientras muestras las dimensiones de la selección:

```
slurp -d
```

- Selecciona un único punto en lugar de una región:

```
slurp -p
```

- Selecciona una salida e imprime su nombre:

```
slurp -o -f '%o'
```

- Selecciona una región determinada y hace una captura de pantalla sin bordes utilizando `grim`:

```
grim -g "$(slurp -w 0)"
```

- Selecciona una región determinada y graba un vídeo sin bordes utilizando `wf-recorder`:

```
wf-recorder --geometry "$(slurp -w 0)"
```

# snake4scores

Muestra las máximas puntuaciones del juego snake4.

Más información: <https://manpages.debian.org/snake4/snake4.6.en.html>.

- Muestra las máximas puntuaciones:

```
snake4scores
```

# sslstrip

Realiza ataques de stripping de Secure Sockets Layer (SSL) basados en el trabajo de Moxie Marlinspike.

Realiza un ataque de suplantación de ARP en conjunto.

Más información: <https://www.kali.org/tools/sslstrip/>.

- Registra sólo el tráfico HTTPS POST en el puerto 10000 por defecto:

```
sslstrip
```

- Registra sólo el tráfico HTTPS POST en el puerto 8080:

```
sslstrip --listen={{8080}}
```

- Registra todo el tráfico SSL hacia y desde el servidor en el puerto 8080:

```
sslstrip --ssl --listen={{8080}}
```

- Registra todo el tráfico SSL y HTTP hacia y desde el servidor en el puerto 8080:

```
sslstrip --listen={{8080}} --all
```

- Especifica la ruta del archivo para almacenar los registros:

```
sslstrip --listen={{8080}} --write={{ruta/a/archivo}}
```

- Muestra la ayuda:

```
sslstrip --help
```

# systemctl

Controla el gestor de sistemas y servicios systemd.

Más información: <https://www.freedesktop.org/software/systemd/man/systemctl.html>.

- Muestra todos los servicios en ejecución:

```
systemctl status
```

- Lista las unidades fallidas:

```
systemctl --failed
```

- Inicia, para, reinicia, recarga o muestra el estado de una unidad:

```
systemctl {{start|stop|restart|reload|status}} {{unidad}}
```

- Habilita o deshabilita una unidad para iniciarla al arrancar:

```
systemctl {{enable|disable}} {{unidad}}
```

- Reinicia systemd, lee unidades nuevas o modificadas:

```
systemctl daemon-reload
```

- Checa si una unidad está activa, habilitada, o en estado fallido:

```
systemctl {{is-active|is-enabled|is-failed}} {{unidad}}
```

- Lista todos los servicios, sockets, unidades auto-montadas filtradas por estado en ejecución o fallido:

```
systemctl list-units --type={{service|socket|automount}} --state={{failed|running}}
```

- Muestra los contenidos y la ruta absoluta del archivo de una unidad:

```
systemctl cat {{unidad}}
```

# systemd-tmpfiles

Crea, elimina y limpia archivos y directorios volátiles y temporales.

Este comando es invocado automáticamente en el arranque por los servicios de systemd, ejecutarlo manualmente tiende a ser innecesario.

Más información: <https://www.freedesktop.org/software/systemd/man/systemd-tmpfiles.html>.

- Crea los archivos y directorios especificados en el archivo de configuración:

```
systemd-tmpfiles --create
```

- Limpia archivos y directorios con los parámetros de edad configurados:

```
systemd-tmpfiles --clean
```

- Elimina archivos y directorios según lo especificado en el archivo de configuración:

```
systemd-tmpfiles --remove
```

- Aplica operaciones en archivos de configuración específicos de usuario:

```
systemd-tmpfiles --create --user
```

- Ejecuta las líneas marcadas para el inicio del arranque:

```
systemd-tmpfiles --create --boot
```



# systool

Vea información de dispositivos del sistema por bus y clases.

Este comando es parte del paquete **sysfs**.

Más información: <https://github.com/linux-ras/sysfsutils>.

- Lista todos los atributos de los dispositivos de un bus (ej. `pci`, `usb`). Vea todos los buses usando `ls /sys/bus`:

```
systool -b {{bus}} -v
```

- Lista todos los atributos de una clase de dispositivos (ej. `drm`, `block`). Vea todas las clases usando `ls /sys/class`:

```
systool -c {{clase}} -v
```

- Muestra solo los controladores de un bus (ej. `pci`, `usb`):

```
systool -b {{bus}} -D
```

# tcpflow

Captura el tráfico TCP para depuración y análisis.

Más información: <https://manned.org/tcpflow>.

- Muestra todos los datos de la interfaz y el puerto indicados:

```
tcpflow -c -i {{eth0}} port {{80}}
```

# tcpkill

Elimina las conexiones TCP en curso especificadas.

Más información: <https://manned.org/tcpkill>.

- Elimina las conexiones en curso de una interfaz, máquina y puerto indicados:

```
tcpkill -i {{eth1}} host {{192.95.4.27}} and port {{2266}}
```

# terraria

Crea e inicia un servidor Terraria sin interfaz gráfica.

Más información: <https://terraria.wiki.gg/wiki/Server>.

- Inicia la configuración interactiva de un servidor:

```
{{ruta/a/TerrariaServer}}
```

- Inicia un servidor Terraria:

```
{{ruta/a/TerrariaServer}} -world {{ruta/a/world.wld}}
```

# thunar

Administrador de archivos gráficos para entornos de escritorio XFCE.

Más información: <https://docs.xfce.org/xfce/thunar/start>.

- Abre una nueva ventana mostrando el directorio actual:

```
thunar
```

- Abre la utilidad de cambio de nombre masivo:

```
thunar --bulk-rename
```

- Cierra todas las ventanas abiertas de thunar:

```
thunar --quit
```

# togglesebool

Voltea los valores actuales (no persistentes) de los booleanos de SELinux.

Nota: Esta herramienta ha quedado obsoleta y a menudo se elimina en favor de **setsebool**.

Más información: <https://manned.org/togglesebool>.

- Cambia los valores actuales (no persistentes) de los booleanos especificados:

```
sudo togglesebool {{virt_use_samba virt_use_usb ...}}
```

# top

Muestra información dinámica en tiempo real sobre procesos ejecutándose.

Más información: <https://manned.org/top>.

- Inicia top:

```
top
```

- Oculta los procesos inactivos o zombies:

```
top -i
```

- Muestra solo procesos pertenecientes a un usuario dado:

```
top -u {{usuario}}
```

- Ordena procesos por una columna:

```
top -o {{nombre_columna}}
```

- Muestra los hilos individuales de un proceso dado:

```
top -Hp {{identificador_de_proceso}}
```

- Muestra solo los procesos con un(os) PID(s) dado(s), separados por comas.  
(Normalmente no se conoce el PID de antemano. Este ejemplo lo obtiene del nombre del proceso):

```
top -p $(pgrep -d ',' {{nombre_proceso}})
```

- Obtén ayuda acerca de los comandos interactivos:

```
?
```

# tor

Habilita la comunicación anónima a través de la red Tor.

Más información: <https://manned.org/tor>.

- Conecta a la red Tor:

```
tor
```

- Vea la configuración de Tor:

```
tor --config
```

- Comprueba el estado de Tor:

```
tor --status
```

- Ejecuta como cliente:

```
tor --client
```

- Ejecuta como relé:

```
tor --relay
```

- Ejecuta como puente:

```
tor --bridge
```

- Ejecuta como servicio oculto:

```
tor --hidden-service
```



# torify

Enruta el tráfico de red a través de la red Tor.

Nota: Este comando está desfasado, y ahora es un empaquetador compatible con versiones anteriores de **torsocks**.

Más información: <https://manned.org/torify>.

- Enruta el tráfico a través de Tor:

```
torify {{comando}}
```

- Activa o desactiva Tor en el intérprete de comandos:

```
torify {{on|off}}
```

- Crea un intérprete de órdenes con Tor activado:

```
torify --shell
```

- Checa si hay un intérprete de órdenes con Tor activado:

```
torify show
```

- Especifica el archivo de configuración de Tor:

```
torify -c {{archivo_de_configuración}} {{comando}}
```

- Usa un proxy Tor SOCKS específico:

```
torify -P {{proxy}} {{comando}}
```

- Redirige la salida a un archivo:

```
torify {{comando}} > {{ruta/a/archivo_de_salida}}
```

# torsocks

Utiliza cualquier aplicación a través de la red de Tor.

Más información: <https://gitlab.torproject.org/tpo/core/torsocks/>.

- Ejecuta un comando usando Tor:

```
torsocks {{comando}}
```

- Activa o desactiva Tor en este intérprete de comandos:

```
. torsocks {{on|off}}
```

# tunelp

Establece varios parámetros para dispositivos de puerto paralelo para la solución de problemas o para un mejor rendimiento.

Parte de **util-linux**.

Más información: <https://manned.org/tunelp>.

- Comprueba el e[s]tado de un dispositivo de puerto paralelo:

```
tunelp --status {{/dev/lp0}}
```

- Restablece un puerto paralelo:

```
tunelp --reset {{/dev/lp0}}
```

- Utiliza un [i]RQ dado para un dispositivo, cada uno representando una línea de interrupción:

```
tunelp -i 5 {{/dev/lp0}}
```

- Intenta imprimir varias veces un [c]arácter en la impresora antes de permanecer inactiva durante un [t]iempo determinado:

```
tunelp --chars {{veces}} --time {{tiempo_en_centisegundos}}  
{{/dev/lp0}}
```

- Activa o desactiva el [a]bortar por error (desactivado por defecto):

```
tunelp --abort {{on|off}}
```

# turbostat

Informa de la topología del procesador, frecuencia, temperatura, potencia y estadísticas de inactividad.

Más información: <https://manned.org/turbostat.8>.

- Muestra las estadísticas cada cinco segundos:

```
sudo turbostat
```

- Muestra las estadísticas cada cierto número de segundos:

```
sudo turbostat -i {{n_segundos}}
```

- Muestra información sin decodificar ni imprimir la cabecera de configuración del sistema:

```
sudo turbostat --quiet
```

- Muestra información útil sobre el CPU cada segundo, sin información de cabecera:

```
sudo turbostat --quiet --interval 1 --cpu 0-  
{{cuenta_hilos_CPU}} --show  
"PkgWatt", "Busy%", "Core", "CoreTmp", "Thermal"
```

- Muestra ayuda:

```
turbostat --help
```

# ubuntu-bug

Este comando es un alias de **apport-bug**.

Más información: <https://manned.org/ubuntu-bug>.

- Muestra la documentación del comando original:

```
tldr apport-bug
```

# ufw

Cortafuegos sin complicaciones (Uncomplicated Firewall).

Interfaz de usuario de **iptables** para facilitar la configuración de un firewall.

Más información: <https://wiki.ubuntu.com/UncomplicatedFirewall>.

- Activa ufw:

```
ufw enable
```

- Desactiva ufw:

```
ufw disable
```

- Muestra reglas del ufw, junto con sus números:

```
ufw status numbered
```

- Permite el tráfico entrante en el puerto 5432 en este host con un comentario que identifique el servicio:

```
ufw allow {{5432}} comment "{{servicio}}"
```

- Permite sólo el tráfico TCP desde 192.168.0.4 a cualquier dirección de este host, en el puerto 22:

```
ufw allow proto {{tcp}} from {{192.168.0.4}} to {{any}} port {{22}}
```

- Deniega tráfico en el puerto 80 en este host:

```
ufw deny {{80}}
```

- Deniega todo el tráfico al puerto 22:

```
ufw deny proto {{udp}} from {{any}} to {{any}} port {{22}}
```

- Elimina una regla concreta. El número de la regla puede obtenerse del comando `ufw status numbered`:

```
ufw delete {{número_de_regla}}
```

# useradd

Crea un usuario nuevo.

Más información: <https://manned.org/useradd>.

- Crea un usuario nuevo:

```
useradd {{nombre}}
```

- Crea un usuario nuevo con el directorio home predeterminado:

```
useradd --create-home {{nombre}}
```

- Crea un usuario nuevo con una shell específica:

```
useradd --shell {{ruta/a/la/shell}} {{nombre}}
```

- Crea un usuario nuevo perteneciente a grupos adicionales (tener en cuenta que no existen espacios en blanco):

```
useradd --groups {{grupo1,grupo2}} {{nombre}}
```

- Crea un usuario nuevo del sistema sin directorio home:

```
useradd --no-create-home --system {{nombre}}
```

# userdel

Elimina una cuenta de usuario o elimina un usuario de un grupo.

Nota: todos los comandos deben ser ejecutados como root.

Más información: <https://manned.org/userdel>.

- Elimina un usuario:

```
userdel {{nombre}}
```

- Elimina un usuario junto con su directorio home y mail spool:

```
userdel --remove {{nombre}}
```

- Elimina un usuario de un grupo:

```
userdel {{nombre}} {{grupo}}
```

- Elimina un usuario en otro directorio root:

```
userdel --root {{ruta/al/otro/root}} {{nombre}}
```



# usermod

Modifica una cuenta de usuario.

Más información: <https://manned.org/usermod>.

- Cambia el nombre de un usuario:

```
usermod -l {{nuevo_nombre}} {{usuario}}
```

- Añade un usuario a grupos suplementarios (tener en cuenta los espacios en blanco):

```
usermod -a -G {{grupo1,grupo2}} {{usuario}}
```

- Crea un nuevo directorio home para un usuario y mueve sus archivos a él:

```
usermod -m -d {{ruta/al/home}} {{usuario}}
```

# vipw

Edita el archivo de contraseñas.

Más información: <https://manned.org/vipw>.

- Edita el archivo de contraseñas:

```
vipw
```

- Muestra la versión actual de vipw:

```
vipw --version
```

# vmstat

Reporta información sobre procesos, memoria, paginación, IO en bloque, traps, discos y actividad del CPU.

Más información: <https://manned.org/vmstat>.

- Muestra las estadísticas de la memoria virtual:

```
vmstat
```

- Muestra informes cada 2 segundos por 5 veces:

```
vmstat {{2}} {{5}}
```

# vnstat

Monitor de tráfico de red de línea de comandos.

Más información: <https://manned.org/vnstat>.

- Muestra un resumen de tráfico de todas las interfaces:

```
vnstat
```

- Muestra un resumen de tráfico de una interfaz de red específica:

```
vnstat -i {{interfaz_de_red}}
```

- Muestra estadísticas en vivo de una interfaz de red específica:

```
vnstat -l -i {{interfaz_de_red}}
```

- Muestra estadísticas de tráfico por hora durante las últimas 24 horas mediante un gráfico de barras:

```
vnstat -hg
```

- Mide y muestra el tráfico promedio por 30 segundos:

```
vnstat -tr {{30}}
```

# vnstati

Genera imágenes PNG compatibles con vnStat.

Más información: <https://manned.org/vnstati>.

- Genera un resumen de los últimos dos meses, días, etc:

```
vnstati --summary --iface {{interfaz_de_red}} --output  
{{ruta/a/salida.png}}
```

- Genera los 10 días con mayor tráfico de todos los tiempos:

```
vnstati --top 10 --iface {{interfaz_de_red}} --output {{ruta/  
a/salida.png}}
```

- Genera estadísticas de tráfico mensual de los últimos 12 meses:

```
vnstati --months --iface {{interfaz_de_red}} --output {{ruta/  
a/salida.png}}
```

- Genera estadísticas de tráfico por hora de las últimas 24 horas:

```
vnstati --hours --iface {{interfaz_de_red}} --output {{ruta/  
a/salida.png}}
```

# W

Muestra quien ha iniciado sesión y sus procesos.

Más información: <https://www.geeksforgeeks.org/w-command-in-linux-with-examples/>.

- Muestra información sobre todos los usuarios que han iniciado sesión actualmente:

```
w
```

- Muestra información sobre un usuario específico:

```
w {{nombre_de_usuario}}
```

- Muestra información sin incluir la cabecera:

```
w --no-header
```

- Muestra información sin incluir las columnas de inicio de sesión, JCPU y PCPU:

```
w --short
```

# waypipe

Ejecuta remotamente aplicaciones gráficas bajo un compositor para Wayland.

Más información: <https://gitlab.freedesktop.org/mstoeckl/waypipe>.

- Ejecuta un programa gráfico remotamente y muéstralo localmente:

```
waypipe ssh {{usuario}}@{{servidor}} {{programa}}
```

- Abre un túnel SSH para ejecutar cualquier programa de forma remota y visualizarlo localmente:

```
waypipe ssh {{usuario}}@{{servidor}}
```

# wmctrl

CLI para X Window Manager.

Más información: <https://manned.org/wmctrl>.

- Lista todas las ventanas, gestionadas por el gestor de ventanas:

```
wmctrl -l
```

- Cambia a la primera ventana cuyo título (parcial) coincida:

```
wmctrl -a {{título_ventana}}
```

- Mueve una ventana al espacio de trabajo actual, levántala y dale foco:

```
wmctrl -R {{título_ventana}}
```

- Cambia a un espacio de trabajo:

```
wmctrl -s {{número_de_espacio_de_trabajo}}
```

- Selecciona una ventana y activa la pantalla completa:

```
wmctrl -r {{título_ventana}} -b toggle,fullscreen
```

- Selecciona una ventana y muévela a un espacio de trabajo:

```
wmctrl -r {{título_ventana}} -t  
{{número_de_espacio_de_trabajo}}
```



# wofi

Un lanzador de aplicaciones para compositores Wayland basados en wlroots, similar a **rofi** y **dmenu**.

Más información: <https://hg.sr.ht/~scoopta/wofi>.

- Muestra la lista de aplicaciones:

```
wofi --show drun
```

- Muestra la lista de todos los comandos:

```
wofi --show run
```

- Envía una lista de elementos a `stdin` e imprime el elemento seleccionado en `stdout`:

```
printf "{{Choice1\nChoice2\nChoice3}}" | wofi --dmenu
```

# xclip

Herramienta para manipular el portapapeles de X11, similar a **xsel**.

Maneja la selección primaria y secundaria de X y el portapapeles (**Ctrl + C**/**Ctrl + V**).

Más información: <https://manned.org/xclip>.

- Copia la salida de un comando a la selección primaria de X11:

```
echo 123 | xclip
```

- Copia la salida de un commando a una selección de X11:

```
echo 123 | xclip -selection {{primary|secondary|clipboard}}
```

- Copia la salida de un commando al portapapeles, usando notación corta:

```
echo 123 | xclip -sel clip
```

- Copia el contenido de un fichero al portapapeles:

```
xclip -sel clip {{archivo.txt}}
```

- Copia el contenido de un fichero con formato PNG al portapapeles:

```
xclip -sel clip -t image/png {{archivo.png}}
```

- Copia la entrada del usuario al portapapeles:

```
xclip -i
```

- Imprime el contenido de la selección primaria de X11:

```
xclip -o
```

- Imprime el contenido del portapapeles:

```
xclip -o -sel clip
```

# xdg-open

Abre un archivo o URL en la aplicación predeterminada del usuario.

Más información: <https://portland.freedesktop.org/doc/xdg-open.html>.

- Abre el directorio actual en el explorador de archivos predeterminado:

```
xdg-open .
```

- Abre una URL en el navegador predeterminado:

```
xdg-open {{https://www.ejemplo.es}}
```

- Abre una imagen en el visor de imágenes predeterminado:

```
xdg-open {{ruta/a_la/imagen}}
```

- Abre un PDF en el visor de PDF predeterminado:

```
xdg-open {{ruta/al/pdf}}
```

- Muestra la ayuda:

```
xdg-open --help
```

# xfreerdp

Implementación libre del protocolo de escritorio remoto (Remote Desktop Protocol).

Más información: <https://www.freerdp.com>.

- Conecta con un servidor FreeRDP:

```
xfreerdp /u:{{nombre_de_usuario}} /p:{{contraseña}} /v:{{direccion_ip}}
```

# xfs\_repair

Repara un sistema de archivos XFS.

Más información: [https://manned.org/xfs\\_repair](https://manned.org/xfs_repair).

- Repara una partición:

```
sudo xfs_repair {{ruta/a/partición}}
```

# ydotool

Controla las entradas de teclado y ratón mediante comandos de forma agnóstica al servidor de visualización.

Más información: <https://github.com/ReimuNotMoe/ydotool>.

- Inicia el daemon ydotool en segundo plano:

```
ydotoold
```

- Realiza un clic con el botón izquierdo:

```
ydotool click 0xC0
```

- Realiza un clic con el botón derecho:

```
ydotool click 0xC1
```

- Ingresa la combinación de teclas Alt+F4:

```
ydotool key 56:1 62:1 62:0 56:0
```

# yum

Administrador de paquetes para RHEL, CentOS y Fedora (para versiones anteriores).

Más información: <https://manned.org/yum>.

- Instala un nuevo paquete:

```
yum install {{paquete}}
```

- Instala un nuevo paquete respondiendo sí a todas las preguntas (también trabaja con actualizaciones, útil para actualizaciones automáticas):

```
yum -y install {{paquete}}
```

- Encuentra que paquete provee un archivo determinado:

```
yum provides {{comando}}
```

- Elimina un paquete:

```
yum remove {{paquete}}
```

- Muestra las actualizaciones disponibles para los paquetes instalados:

```
yum check-update
```

- Actualiza los paquetes instalados a las versiones más recientes disponibles:

```
yum upgrade
```

# zramctl

Configura y controla dispositivos zram.

Usa **mkfs** o **mkswap** para formatear dispositivos zram a particiones.

Más información: <https://manned.org/zramctl>.

- Comprueba si zram está habilitado:

```
lsmod | grep -i zram
```

- Habilita zram con un número dinámico de dispositivos (usa `zramctl` para configurar más dispositivos):

```
sudo modprobe zram
```

- Habilita zram con exactamente 2 dispositivos:

```
sudo modprobe zram num_devices={{2}}
```

- Encuentra e inicializa el siguiente dispositivo zram libre a una unidad virtual de 2 GB usando compresión LZ4:

```
sudo zramctl --find --size {{2GB}} --algorithm {{lz4}}
```

- Lista los dispositivos actualmente inicializados:

```
sudo zramctl
```



# zypper

Utilidad para la gestión de paquetes en SUSE y openSUSE.

Más información: [https://en.opensuse.org/SDB:Zypper\\_manual](https://en.opensuse.org/SDB:Zypper_manual).

- Sincroniza la lista de paquetes y versiones disponibles:

```
zypper refresh
```

- Instala un nuevo paquete:

```
zypper install {{paquete}}
```

- Elimina un paquete:

```
zypper remove {{paquete}}
```

- Actualiza los paquetes instalados a la versión más reciente disponible:

```
zypper update
```

- Busca en los repositorios un paquete mediante una palabra clave:

```
zypper search {{palabra_clave}}
```

- Muestra información relacionada con los repositorios configurados:

```
zypper repos --sort-by-priority
```

Netbsd

# df

Muestra una visión general del uso del espacio en disco del sistema de archivos.

Más información: <https://man.netbsd.org/df.1>.

- Muestra todos los sistemas de ficheros y su uso de disco usando unidades de 512 bytes:

```
df
```

- Utiliza palabras [h]umanas para indicar unidades (basadas en potencias de 1024):

```
df -h
```

- Muestra todos los campos de la(s) estructura(s) devuelta(s) por `statvfs`:

```
df -G
```

- Muestra el sistema de archivos y su uso del disco que contiene el archivo o directorio dado:

```
df {{ruta/al/archivo_o_directorio}}
```

- Incluye estadísticas sobre el número de nodos-[i] libres y utilizados:

```
df -i
```

- Utiliza unidades de 1024 bytes al escribir las cifras de espacio:

```
df -k
```

- Muestra la información de manera [P]ortable:

```
df -P
```

# pkgin

Gestiona paquetes binarios **pkgsrc** en NetBSD.

Más información: <https://pkgin.net/#usage>.

- Instala un paquete:

```
pkgin install {{paquete}}
```

- Elimina un paquete y sus dependencias:

```
pkgin remove {{paquete}}
```

- Actualiza todos los paquetes:

```
pkgin full-upgrade
```

- Busca un paquete:

```
pkgin search {{palabra_clave}}
```

- Lista los paquetes instalados:

```
pkgin list
```

- Elimina dependencias innecesarias:

```
pkgin autoremove
```

# sed

Edita texto de manera programable.

Vea también: **awk**, **ed**.

Más información: <https://man.netbsd.org/sed.1>.

- Reemplaza todas las ocurrencias de `apple` (expresión regular básica) por `mango` (expresión regular básica) en todas las líneas de entrada e imprime el resultado en `stdout`:

```
{{comando}} | sed 's/apple/mango/g'
```

- Ejecuta un [f]ichero con un script e imprime el resultado en `stdout`:

```
{{comando}} | sed -f {{ruta/al/script.sed}}
```

- Retrasa abrir cada archivo hasta que se aplique a una línea de entrada un comando que contenga la función `w` u otra similar:

```
{{comando}} | sed -fa {{ruta/al/script.sed}}
```

- Activa la extensión de GNU de expresiones re[g]ulares:

```
{{comando}} | sed -fg {{ruta/al/script.sed}}
```

- Sustituye todas las ocurrencias de `apple` (expresión regular extendida) por `APPLE` (expresión regular extendida) en todas las líneas de entrada e imprime el resultado en `stdout`:

```
{{comando}} | sed -E 's/(apple)/\U\1/g'
```

- Imprime solo la primera línea en `stdout`:

```
{{comando}} | sed -n '1p'
```

- Sustituye todas las apariciones de `apple` (expresión regular básica) por `mango` (expresión regular básica) en un archivo específico y sobrescribe el archivo original en su lugar:

```
sed -i 's/apple/mango/g' {{ruta/al/archivo}}
```

# sockstat

Lista sockets abiertos de Internet o dominios UNIX.

Nota: este programa es una reescritura para NetBSD 3.0 de **sockstat** de FreeBSD.

Vea también: **netstat**.

Más información: <https://man.netbsd.org/sockstat.1>.

- Muestra información de sockets IPv4, IPv6 y Unix que estén escuchando y conectados:

```
sockstat
```

- Muestra información para sockets IPv[4]/IPv[6] escuchando ([l]istening) sobre [p]uertos específicos usando un [P]rotocolo específico:

```
sockstat -{{4|6}} -l -P {{tcp|udp|sctp|divert}} -p  
{{puerto1,puerto2...}}
```

- También muestra sockets [c]onectados, mostrando sockets [u]nix:

```
sockstat -cu
```

- Solo muestra salida [n]umérica, sin resolver nombres simbólicos para direcciones y puertos:

```
sockstat -n
```

- Lista sockets de una [f]amilia de direcciones específica:

```
sockstat -f {{inet|inet6|local|unix}}
```

Osx

# aa

Este comando es un alias de **yaa**.

- Muestra la documentación del comando original:

```
tldr yaa
```



# afinfo

Analizador de metadatos de archivos de audio para OS X.

Comando nativo de OS X.

Más información: <https://keith.github.io/xcode-man-pages/afinfo.1.html>.

- Muestra información de un archivo de audio dado:

```
afinfo {{ruta/al/archivo}}
```

- Muestra una descripción de una línea del archivo de audio:

```
afinfo --brief {{ruta/al/archivo}}
```

- Muestra información de metadatos y contenido del InfoDictionary del archivo de audio:

```
afinfo --info {{ruta/al/archivo}}
```

- Imprime la salida en formato XML:

```
afinfo --xml {{ruta/al/archivo}}
```

- Muestra advertencias para el archivo de audio, si las hubiera:

```
afinfo --warnings {{ruta/al/archivo}}
```

- Muestra ayuda para un uso completo:

```
afinfo --help
```

# afplay

Reproductor de audio de línea de comandos.

Más información: <https://keith.github.io/xcode-man-pages/afplay.1.html>.

- Reproduce un archivo de audio (espera hasta que finalice la reproducción):

```
afplay {{ruta/al/archivo}}
```

- Reproduce un archivo de audio a una velocidad 2x (velocidad de reproducción):

```
afplay --rate {{2}} {{ruta/al/archivo}}
```

- Reproduce un archivo de audio a la mitad de velocidad:

```
afplay --rate {{0.5}} {{ruta/al/archivo}}
```

- Reproduce los primeros N segundos de un archivo de audio:

```
afplay --time {{segundos}} {{ruta/al/archivo}}
```

# aiac

Utiliza OpenAI para generar configuraciones IaC, utilidades, consultas y más.

Más información: <https://github.com/gofireflyio/aiac>.

- Genera Terraform para una cuenta de almacenamiento Azure:

```
aiac get terraform {{for an azure storage account}}
```

- Genera un Dockerfile para nginx:

```
aiac get dockerfile {{for a secured nginx}}
```

- Genera una GitHub action que aplica Terraform:

```
aiac get github action {{that plans and applies terraform}}
```

- Genera un escáner de puertos en Python:

```
aiac get python {{code that scans all open ports in my network}}
```

- Genera una consulta MongoDB:

```
aiac get mongo {{query that aggregates all documents by created date}}
```

# airport

Utilidad de configuración de red inalámbrica.

Más información: <https://keith.github.io/xcode-man-pages/airport.1.html>.

- Muestra la información del estado actual de la red inalámbrica:

```
airport --getinfo
```

- Detecta tráfico inalámbrico en el canal 1:

```
airport sniff {{1}}
```

- Busca redes inalámbricas disponibles:

```
airport --scan
```

- Desasociarse de la red actual:

```
sudo airport --disassociate
```

# airportd

Gestiona las interfaces inalámbricas.

No debe invocarse manualmente.

Más información: <https://keith.github.io/xcode-man-pages/airportd.8.html>.

- Inicia el daemon:

```
airportd
```

# apachectl

Interfaz de control de Apache HTTP Server para macOS.

Más información: <https://keith.github.io/xcode-man-pages/apachectl.8.html>.

- Inicia la tarea launchd org.apache.httpd:

```
apachectl start
```

- Finaliza la tarea launchd:

```
apachectl stop
```

- Finaliza e inicia la tarea launchd:

```
apachectl restart
```

# applecamerad

Gestor de cámara.

No debe ser invocado manualmente.

Más información: <https://www.theiphonewiki.com/wiki/Services>.

- Inicia el proceso residente:

`applecamerad`

# appsleefd

Proporciona servicios de suspensión de aplicaciones.

No debe invocarse manualmente.

Más información: <https://keith.github.io/xcode-man-pages/appsleefd.8.html>.

- Inicia el daemon:

```
appsleefd
```



# arch

Muestra el nombre de la arquitectura del sistema, o ejecuta un comando bajo una arquitectura diferente.

Vea también: **uname**.

Más información: <https://keith.github.io/xcode-man-pages/arch.1.html>.

- Muestra la arquitectura del sistema:

```
arch
```

- Ejecuta un comando usando x86\_64:

```
arch -x86_64 "{{comando}}"
```

- Ejecuta un comando usando arm:

```
arch -arm64 "{{comando}}"
```

# archey

Herramienta sencilla para mostrar información del sistema con estilo.

Más información: <https://github.com/joshfinnie/archey-osx>.

- Muestra información del sistema:

```
archey
```

- Muestra información del sistema sin colorear la salida:

```
archey --nocolor
```

- Muestra información del sistema, usando MacPorts en lugar de Homebrew:

```
archey --macports
```

- Muestra información del sistema sin verificación dirección IP:

```
archey --offline
```

# as

Ensamblador portable GNU.

Principalmente destinado a ensamblar la salida de **gcc** para ser utilizada por **ld**.

Más información: <https://keith.github.io/xcode-man-pages/as.1.html>.

- Ensambla un archivo, escribiendo la salida en `a.out`:

```
as {{ruta/al/archivo.s}}
```

- Ensambla la salida a un archivo especificado:

```
as {{ruta/al/archivo.s}} -o {{salida.o}}
```

- Genera resultados más rápidos omitiendo los espacios en blanco y el preprocesamiento de comentarios. (Solo debe usarse para compiladores de confianza):

```
as -f {{ruta/al/archivo.s}}
```

- Incluye una ruta determinada a la lista de directorios para buscar archivos especificados en las directivas `.include`:

```
as -I {{ruta/al/directorio}} {{ruta/al/archivo.s}}
```

# asr

Restaura (copia) una imagen de disco en un volumen.

El nombre del comando significa Restauración de Software de Apple.

Más información: <https://keith.github.io/xcode-man-pages/asr.8.html>.

- Restaura una imagen de disco en un volumen:

```
sudo asr restore --source {{nombre_de_imagen.dmg}} --target  
{{ruta/al/volumen}}
```

- Borra el volumen deseado antes de restaurar:

```
sudo asr restore --source {{nombre_de_imagen.dmg}} --target  
{{ruta/al/volumen}} --erase
```

- Omite la verificación después de restaurar:

```
sudo asr restore --source {{nombre_de_imagen.dmg}} --target  
{{ruta/al/volumen}} --noverify
```

- Clona volúmenes sin el uso de una imagen de disco intermedia:

```
sudo asr restore --source {{ruta/al/volumen}} --target  
{{ruta/al/volumen_clonado}}
```

# autofs

Ejecuta **automount** al inicio del sistema y en cambios en la configuración de red.

No debe ser invocado manualmente.

Más información: <https://keith.github.io/xcode-man-pages/autofs.8.html>.

- Inicia el proceso residente:

`autofs`

# automount

Lee el archivo `/etc/auto_master` y monta **autofs** en los puntos de montaje apropiados para activar el montaje bajo demanda de directorios. Esencialmente, es una forma de iniciar manualmente el proceso de automontaje del sistema.

Nota: Lo más probable es que necesites ejecutarlo con **sudo** si no tienes los permisos necesarios.

Más información: <https://keith.github.io/xcode-man-pages/automount.8.html>.

- Ejecuta automount, vacía la caché(-c) de antemano, y es detallista(-v) al respecto (uso más común):

```
automount -cv
```

- Desmonta automáticamente transcurridos 5 minutos (300 segundos) de inactividad:

```
automount -t 300
```

- Desmonta cualquier cosa previamente montada por automount y/o definida en `/etc/auto_master`:

```
automount -u
```

# automountd

Un daemon de montaje/desmontaje automático para **autofs**. Iniciado bajo demanda por **launchd**.

No debe ser invocado manualmente.

Más información: <https://keith.github.io/xcode-man-pages/automountd.8.html>.

- Inicia el daemon:

```
automountd
```

- Registra más detalles en `syslog`:

```
automountd -v
```

# auvaltool

Herramienta de validación AudioUnit para Mac.

Más información: <https://keith.github.io/xcode-man-pages/auvaltool.1.html>.

- Lista todas las AudioUnits disponibles de cualquier tipo:

```
auvaltool -a
```

- Lista todas las AudioUnits [a]vailable de cualquier tipo con su [l]ocalización:

```
auvaltool -al
```



# avbdeviced

Un servicio para gestionar dispositivos Audio Video Bridging (AVB).

No debe invocarse manualmente.

Más información: <https://keith.github.io/xcode-man-pages/avbdeviced.1.html>.

- Inicia el daemon:

```
avbdeviced
```

# backupd

Crea copias de seguridad de Time Machine y gestiona el historial de copias de seguridad.

No debe invocarse manualmente.

Más información: <https://keith.github.io/xcode-man-pages/backupd.8.html>.

- Inicia el daemon:

```
backupd
```

# base64

Codifica y decodifica usando la representación Base64.

Más información: <https://keith.github.io/xcode-man-pages/base64.1.html>.

- Codifica un archivo:

```
base64 --input={{archivo_plano}}
```

- Decodifica un archivo:

```
base64 --decode --input={{base64_archivo}}
```

- Codifica desde `stdin`:

```
echo -n "{{texto_plano}}" | base64
```

- Decodifica desde `stdin`:

```
echo -n {{base64_texto}} | base64 --decode
```

# bc

Un lenguaje de calculadora de precisión arbitraria.

Vea también: **dc**.

Más información: <https://keith.github.io/xcode-man-pages/bc.1.html>.

- Inicia una sesión interactiva:

```
bc
```

- Inicia una sesión interactiva con la biblioteca mathlib estándar activada:

```
bc --mathlib
```

- Calcula una expresión:

```
bc --expression='{{5 / 3}}'
```

- Ejecuta un script:

```
bc {{ruta/al/script.bc}}
```

- Calcula una expresión con la escala especificada:

```
bc --expression='scale = {{10}}; {{5 / 3}}'
```

- Calcula una función seno/coseno/arctangente/logaritmo natural/exponencial utilizando `mathlib`:

```
bc --mathlib --expression='{{s|c|a|l|e}}({{1}})'
```

# biomesyncd

Sincroniza datos entre dispositivos registrados en la misma cuenta.

No debe invocarse manualmente.

Más información: <https://keith.github.io/xcode-man-pages/biomesyncd.8.html>.

- Inicia el daemon:

```
biomesyncd
```

# biometrickitd

Proporciona soporte para operaciones biométricas.

No debe ser invocado manualmente.

Más información: <https://keith.github.io/xcode-man-pages/biometrickitd.8.html>.

- Inicia el proceso residente:

```
biometrickitd
```

# bird

Esto admite la sincronización de iCloud e iCloud Drive.

No debe ser invocado manualmente.

Más información: <https://keith.github.io/xcode-man-pages/bird.8.html>.

- Inicia el proceso residente:

```
bird
```

# bless

Establece la capacidad de arranque del volumen y las opciones del disco de arranque.

Más información: <https://keith.github.io/xcode-man-pages/bless.8.html>.

- Bendice un volumen sólo con Mac OS X o Darwin, y crea los archivos BootX y `boot.efi` según sea necesario:

```
bless --folder {{/Volumes/Mac OS X/System/Library/
CoreServices}} --bootinfo --bootefi
```

- Configura un volumen que contenga Mac OS 9 y Mac OS X para que sea el volumen activo:

```
bless --mount {{Volumes/Mac OS}} --setBoot
```

- Configura el sistema como NetBoot y localiza un servidor disponible:

```
bless --netboot --server {{bsdp://255.255.255.255}}
```

- Recopila información sobre el volumen seleccionado actualmente (según lo determinado por el firmware), adecuado para la canalización a un programa capaz de analizar las listas de propiedades:

```
bless --info --plist
```



# bnepd

Un servicio que gestiona todas las conexiones de red Bluetooth.

No debe invocarse manualmente.

Más información: <https://keith.github.io/xcode-man-pages/bnepd.8.html>.

- Inicia el daemon:

```
bnepd
```

# caffeinate

Evita que macOS se duerma.

Más información: <https://keith.github.io/xcode-man-pages/caffeinate.8.html>.

- Evita que macOS entre en reposo durante 1 hora (3600 segundos):

```
caffeinate -u -t {{3600}}
```

- Evita que entre en reposo hasta que termine de ejecutarse un comando:

```
caffeinate -s "{{comando}}"
```

- Evita que el sistema entre en reposo hasta que finalice un proceso con el PID especificado:

```
caffeinate -w {{pid}}
```

- Evita que entre en reposo (usa Ctrl + C para salir):

```
caffeinate -i
```

- Evita que el disco entre en reposo (usa Ctrl + C para salir):

```
caffeinate -m
```

# coreaudiod

Servicio para Core Audio, el sistema de audio de Apple.

No debe invocarse manualmente.

Más información: <https://developer.apple.com/library/archive/documentation/MusicAudio/Conceptual/CoreAudioOverview/WhatIsCoreAudio/WhatIsCoreAudio.html>.

- Inicia el daemon:

```
coreaudiod
```

# corebrightnessd

Gestiona Night Shift.

No debe ser invocado manualmente.

Más información: <https://keith.github.io/xcode-man-pages/corebrightnessd.8.html>.

- Inicia el proceso residente:

```
corebrightnessd
```

# coredatad

Programa operaciones CloudKit para clientes de NSPersistentCloudKitContainer.

No debe ser invocado manualmente.

Más información: <https://keith.github.io/xcode-man-pages/coredatad.8.html>.

- Inicia el proceso residente:

`coredatad`

# cot

El editor de texto simple para macOS.

Más información: <https://coteditor.com/>.

- Inicia CotEditor:

```
cot
```

- Abre archivos específicos:

```
cot {{ruta/al/archivo1 ruta/al/archivo2 ...}}
```

- Abre un nuevo documento en blanco:

```
cot --new
```

- Abre un archivo específico y bloquea el terminal hasta que se cierre:

```
cot --wait {{ruta/al/archivo}}
```

- Abre un archivo específico con el cursor en una línea y columna específicas:

```
cot --line {{1}} --column {{80}} {{ruta/al/archivo}}
```

# csrutil

Gestiona la configuración de la Protección de Integridad del Sistema.

Más información: <https://keith.github.io/xcode-man-pages/csrutil.8.html>.

- Muestra el estado de la Protección de Integridad del Sistema:

```
csrutil status
```

- Desactiva la Protección de Integridad del Sistema:

```
csrutil disable
```

- Activa la Protección de Integridad del Sistema:

```
csrutil enable
```

- Muestra la lista de fuentes NetBoot permitidas:

```
csrutil netboot list
```

- Añade una dirección IPv4 a la lista de fuentes NetBoot permitidas:

```
csrutil netboot add {{ip}}
```

- Restablece el estado de Protección de integridad del Sistema y borra la lista NetBoot:

```
csrutil clear
```

# csshX

Herramienta SSH de clúster para macOS.

Más información: <https://github.com/brockgr/csshx>.

- Conectarse a múltiples hosts:

```
csshX {{nombrehost1}} {{nombrehost2}}
```

- Conectarse a múltiples hosts con una clave SSH dada:

```
csshX {{usuario@nombrehost1}} {{usuario@nombrehost2}} --  
ssh_args "-i {{ruta/al/archivo_de_clave.pem}}"
```

- Conectarse a un clúster predefinido desde `/etc/clusters`:

```
csshX cluster1
```



# ctkd

Daemon de SmartCard.

No debe invocarse manualmente.

Más información: <https://keith.github.io/xcode-man-pages/ctkd.8.html>.

- Inicia el daemon:

```
ctkd
```

# cut

Corta campos sean **stdin** o archivos.

Más información: <https://keith.github.io/xcode-man-pages/cut.1.html>.

- Imprime un rango específico de caracteres/campos de cada línea:

```
{{command}} | cut -{{c|f}} {{1|1,10|1-10|1-|-10}}
```

- Imprime un rango de campos de cada línea con un delimitador específico:

```
{{command}} | cut -d "{{{,}}}" -f {{1}}
```

- Imprime un rango de caracteres de cada línea de un archivo específico:

```
cut -c {{1}} {{ruta/al/archivo}}
```

# dark-mode

Controla el modo oscuro de macOS desde la línea de comandos.

Más información: <https://github.com/sindresorhus/dark-mode>.

- Alterna el modo oscuro (lo activa si actualmente está desactivado, lo desactiva si actualmente está activado):

```
dark-mode
```

- Activa el modo oscuro:

```
dark-mode on
```

- Desactiva el modo oscuro:

```
dark-mode off
```

- Verifica si el modo oscuro está activado:

```
dark-mode status
```

# dd

Convierte y copia un archivo.

Más información: <https://keith.github.io/xcode-man-pages/dd.1.html>.

- Crea una unidad USB de arranque desde un archivo isohybrid (como `archlinux-xxx.iso`) y muestra el progreso:

```
dd if={{ruta/al/archivo.iso}} of={{/dev/dispositivo_usb}}  
status=progress
```

- Clona una unidad a otra unidad con un bloque de 4 MB, ignora el error y muestra el progreso:

```
dd bs=4m conv=noerror if={{/dev/dispositivo_de origen}}  
of={{/dev/dispositivo_de destino}} status=progress
```

- Genera un archivo con un número específico de bytes aleatorios utilizando el controlador aleatorio del núcleo:

```
dd bs={{100}} count={{1}} if=/dev/urandom of={{ruta/al/  
archivo_aleatorio}}
```

- Compara el rendimiento de escritura de un disco:

```
dd bs={{1024}} count={{1000000}} if=/dev/zero of={{ruta/para/  
archivo_1GB}}
```

- Genera una copia de seguridad del sistema en un archivo IMG y muestra el progreso:

```
dd if={{/dev/dispositivo_unidad}} of={{ruta/al/archivo.img}}  
status=progress
```

- Comprueba el progreso de una operación dd en curso (ejecuta este comando desde otro intérprete de comandos):

```
kill -USR1 $(pgrep ^dd)
```

# dhcp6d

Servidor DHCPv6 sin estado. Ver también: **InternetSharing**.

No debe invocarse manualmente.

Más información: <https://www.manpagez.com/man/8/dhcp6d/>.

- Inicia el daemon:

```
dhcp6d
```

- Utiliza una configuración personalizada:

```
dhcp6d {{ruta/al/archivo_config}}
```

# diskutil partitionDisk

Utilidad para gestionar particiones en discos y volúmenes.

Parte de **diskutil**.

APM solo es compatible con macOS, MBR está optimizado para DOS, mientras que GPT es compatible con la mayoría de los sistemas operativos modernos.

Más información: <https://keith.github.io/xcode-man-pages/diskutil.8.html>.

- Reformatea un volumen usando el esquema de particionado APM/MBR/GPT, sin dejar particiones en su interior (esto borrará todos los datos del volumen):

```
diskutil partitionDisk {{/dev/dispositivo_de_disco}} 0 {{APM|MBR|GPT}}
```

- Reformatea un volumen, luego crea una única partición usando un sistema de archivos específico llenando todo el espacio libre:

```
diskutil partitionDisk {{/dev/dispositivo_de_disco}} 1 {{APM|MBR|GPT}} {{partición_con_sistema_de_archivos}} {{nombre_de_partición}}
```

- Reformatea un volumen, luego crea una única partición con un sistema de archivos y tamaño específico (ej. 16G para 16GB o 50% para llenar la mitad del tamaño total del volumen):

```
diskutil partitionDisk {{/dev/dispositivo_de_disco}} 1 {{APM|MBR|GPT}} {{partición_con_sistema_de_archivos}} {{nombre_de_partición}} {{tamaño_de_partición}}
```

- Reformatea un volumen y crea múltiples particiones:

```
diskutil partitionDisk {{/dev/dispositivo_disco}} {{número_particiones}} {{APM|MBR|GPT}} {{partición_sistema_archivos1}} {{nombre_partición1}} {{tamaño_partición1}} {{partición_sistema_archivos2}} {{nombre_partición2}} {{tamaño_partición2}} ...
```

- Lista todos los sistemas de ficheros soportados para particionar:

```
diskutil listFilesystems
```

# diskutil

Utilidad para gestionar discos y volúmenes locales.

Más información: <https://keith.github.io/xcode-man-pages/diskutil.8.html>.

- Lista todos los discos, particiones y volúmenes montados actualmente disponibles:

```
diskutil list
```

- Repara las estructuras de datos del sistema de archivos de un volumen:

```
diskutil repairVolume {{/dev/disk_device}}
```

- Desmonta un volumen:

```
diskutil unmountDisk {{/dev/disk_device}}
```

- Expulsa un CD/DVD (primero lo desmonta):

```
diskutil eject {{/dev/disk_device1}}
```

# dockutil

Gestiona los elementos del dock de macOS.

Más información: <https://github.com/kcrawford/dockutil>.

- Añade una aplicación al final del dock del usuario actual:

```
dockutil --add {{ruta/a/la/aplicación}}
```

- Reemplaza una aplicación por otra en el dock del usuario actual:

```
dockutil --add {{ruta/a/la/aplicación}} --replacing  
'{{etiqueta_de_elemento_del_dock}}'
```

- Añade un directorio con opciones de visualización y lo muestra como un icono de carpeta o pila:

```
dockutil --add {{/ruta/al/directorio}} --view {{grill|fan|  
list|auto}} --display {{folder|stack}}
```

- Añade la URL de un elemento del dock después de otro elemento:

```
dockutil --add {{vnc://ejemplo_servidor.local}} --label  
'{{VNC de ejemplo}}' --after  
{{etiqueta_de_elemento_del_dock}}
```

- Elimina una aplicación del dock dado su nombre de etiqueta en el dock:

```
dockutil --remove '{{etiqueta_de_elemento_del_dock}}'
```

- Añade un espaciador en una sección después de una aplicación:

```
dockutil --add '' --type {{spacer|small-spacer|flex-spacer}}  
--section {{apps}} --after {{etiqueta_de_elemento_del_dock}}
```

- Elimina todos los elementos espaciadores:

```
dockutil --remove spacer-tiles
```



# dot\_clean

Fusiona los archivos `._*` con los archivos nativos correspondientes.

Más información: [https://keith.github.io/xcode-man-pages/dot\\_clean.1.html](https://keith.github.io/xcode-man-pages/dot_clean.1.html).

- Fusiona todos los ficheros `._*` recursivamente:

```
dot_clean {{ruta/al/directorio}}
```

- Fusiona todos los `._*` en un directorio sin leer subdirectorios (fusión plana):

```
dot_clean -f {{ruta/al/directorio}}
```

- Fusiona y elimina todos los archivos `._*`:

```
dot_clean -m {{ruta/al/directorio}}
```

- Elimina sólo los archivos `._*` si hay un archivo nativo coincidente:

```
dot_clean -n {{ruta/al/directorio}}
```

- Sigue los enlaces simbólicos:

```
dot_clean -s {{ruta/al/directorio}}
```

- Muestra resultados detallados:

```
dot_clean -v {{ruta/al/directorio}}
```

# du

Uso del disco: estima y resume el uso del espacio de archivos y directorios.

Más información: <https://keith.github.io/xcode-man-pages/du.1.html>.

- Lista los tamaños de un directorio y de cualquier subdirectorio, en la unidad dada (KiB/MiB/GiB):

```
du -{{k|m|g}} {{ruta/al/directorio}}
```

- Enumera los tamaños de un directorio y sus subdirectorios, de forma legible (es decir, seleccionando automáticamente la unidad adecuada para cada tamaño):

```
du -h {{ruta/al/directorio}}
```

- Muestra el tamaño de un único directorio, en unidades legibles para el ser humano:

```
du -sh {{ruta/al/directorio}}
```

- Muestra los tamaños legibles de un directorio y de todos los archivos y directorios que contiene:

```
du -ah {{ruta/al/directorio}}
```

- Lista los tamaños legibles de un directorio y sus subdirectorios, hasta N niveles de profundidad:

```
du -h -d {{2}} {{ruta/al/directorio}}
```

- Lista el tamaño legible de todos los archivos `.jpg` en los subdirectorios del directorio actual y muestra un total acumulado al final:

```
du -ch {{*/*.jpg}}
```

# ed

El editor de texto original de Unix.

Vea también: **awk**, **sed**.

Más información: [https://www.gnu.org/software/ed/manual/ed\\_manual.html](https://www.gnu.org/software/ed/manual/ed_manual.html).

- Inicia una sesión de edición interactiva con un documento vacío:

```
ed
```

- Inicia una sesión de editor interactivo con un documento vacío y un [p]rompt específico:

```
ed -p '> '
```

- Inicia una sesión de editor interactivo con un documento vacío y sin diagnósticos, recuento de bytes y prompt '!':

```
ed -s
```

- Edita un archivo específico (muestra el recuento de bytes del archivo cargado):

```
ed {{ruta/al/archivo}}
```

- Reemplaza una cadena con un reemplazo específico para todas las líneas:

```
,s/{{expresión_reguar}}/{{reemplazo}}/g
```

# emond

Servicio de Monitor de Eventos que acepta eventos de varios servicios, los ejecuta a través de un simple motor de reglas, y toma acciones.

Las acciones pueden ejecutar comandos, enviar correos electrónicos o mensajes SMS.

Más información: <https://keith.github.io/xcode-man-pages/emond.8.html>.

- Inicia el daemon:

```
emond
```

- Especifica las reglas que emond debe procesar indicando una ruta a un archivo o directorio:

```
emond -r {{ruta/al/archivo_o_directorio}}
```

- Utiliza un archivo de configuración específico:

```
emond -c {{ruta/al/archivo_de_configuración}}
```

# fdesetup

Establece y recupera información relacionada con FileVault.

Más información: <https://keith.github.io/xcode-man-pages/fdesetup.8.html>.

- Lista los usuarios actuales habilitados para FileVault:

```
sudo fdesetup list
```

- Obtén el estado actual de FileVault:

```
fdesetup status
```

- Añade un usuario habilitado para FileVault:

```
sudo fdesetup add -usertoadd {{usuario1}}
```

- Habilita FileVault:

```
sudo fdesetup enable
```

- Desactiva FileVault:

```
sudo fdesetup disable
```

# fsck

Comprueba la integridad de un sistema de archivos o los repara. El sistema de ficheros debe estar desmontado en el momento de ejecutar el comando.

Es una envoltura que llama a **fsck\_hfs**, **fsck\_apfs**, **fsck\_msdos**, **fsck\_exfat**, y **fsck\_udf** según sea necesario.

Más información: <https://keith.github.io/xcode-man-pages/fsck.8.html>.

- Comprueba el sistema de ficheros `/dev/sdX`, informando de cualquier bloque dañado:

```
fsck {{/dev/sdX}}
```

- Comprueba el sistema de ficheros `/dev/sdX` sólo si está limpio, informando de los bloques dañados y dejando que el usuario elija interactivamente si repara cada uno de ellos:

```
fsck -f {{/dev/sdX}}
```

- Comprueba el sistema de archivos `/dev/sdX` sólo si está limpio, informando de los bloques dañados y reparándolos automáticamente:

```
fsck -fy {{/dev/sdX}}
```

- Comprueba el sistema de archivos `/dev/sdX`, informando si se ha desmontado correctamente:

```
fsck -q {{/dev/sdX}}
```

# ftxdiff

Compara las diferencias entre dos fuentes.

Más información: <https://developer.apple.com/fonts>.

- Envía las diferencias a un archivo de texto específico:

```
ftxdiff --output {{ruta/al/archivo_de_fontdif.txt}} {{ruta/al/archivo_ont_1.ttc}} {{ruta/al/archivo_font_2.ttc}}
```

- Incluye nombres de glifos en la salida:

```
ftxdiff --include-glyph-names
```

- Incluye nombres unicode en la salida:

```
ftxdiff --include-unicode-names
```

# g[

Este comando es un alias de **-p linux** [.

- Muestra la documentación del comando original:

```
tldr -p linux [
```



# gawk

Este comando es un alias de **-p linux awk**.

- Muestra la documentación del comando original:

```
tldr -p linux awk
```

# gb2sum

Este comando es un alias de **-p linux b2sum**.

- Muestra la documentación del comando original:

```
tldr -p linux b2sum
```

# gbase32

Este comando es un alias de **-p linux base32**.

- Muestra la documentación del comando original:

```
tldr -p linux base32
```

# gbase64

Este comando es un alias de **-p linux base64**.

- Muestra la documentación del comando original:

```
tldr -p linux base64
```

# gbasename

Este comando es un alias de **-p linux basename**.

- Muestra la documentación del comando original:

```
tldr -p linux basename
```

# gbasenc

Este comando es un alias de **-p linux basenc**.

- Muestra la documentación del comando original:

```
tldr -p linux basenc
```

# gcat

Este comando es un alias de **-p linux cat**.

- Muestra la documentación del comando original:

```
tldr -p linux cat
```

# gchcon

Este comando es un alias de **-p linux chcon**.

- Muestra la documentación del comando original:

```
tldr -p linux chcon
```



# gchgrp

Este comando es un alias de **-p linux chgrp**.

- Muestra la documentación del comando original:

```
tldr -p linux chgrp
```

# gchmod

Este comando es un alias de **-p linux chmod**.

- Muestra la documentación del comando original:

```
tldr -p linux chmod
```

# gchown

Este comando es un alias de **-p linux chown**.

- Muestra la documentación del comando original:

```
tldr -p linux chown
```

# gchroot

Este comando es un alias de **-p linux chroot**.

- Muestra la documentación del comando original:

```
tldr -p linux chroot
```

# gcksum

Este comando es un alias de **-p linux cksum**.

- Muestra la documentación del comando original:

```
tldr -p linux cksum
```

# gcomm

Este comando es un alias de **-p linux comm**.

- Muestra la documentación del comando original:

```
tldr -p linux comm
```

# gcp

Este comando es un alias de **-p linux cp**.

- Muestra la documentación del comando original:

```
tldr -p linux cp
```

# gcsplit

Este comando es un alias de **-p linux csplit**.

- Muestra la documentación del comando original:

```
tldr -p linux csplit
```



# gcut

Este comando es un alias de **-p linux cut**.

- Muestra la documentación del comando original:

```
tldr -p linux cut
```

# gdate

Este comando es un alias de **-p linux date**.

- Muestra la documentación del comando original:

```
tldr -p linux date
```

# gdd

Este comando es un alias de **-p linux dd**.

- Muestra la documentación del comando original:

```
tldr -p linux dd
```

# gdf

Este comando es un alias de **-p linux df**.

- Muestra la documentación del comando original:

```
tldr -p linux df
```

# gdir

Este comando es un alias de **-p linux dir**.

- Muestra la documentación del comando original:

```
tldr -p linux dir
```

# gdircolors

Este comando es un alias de **-p linux dircolors**.

- Muestra la documentación del comando original:

```
tldr -p linux dircolors
```

# gdirname

Este comando es un alias de **-p linux dirname**.

- Muestra la documentación del comando original:

```
tldr -p linux dirname
```

# gdnsdomainname

Este comando es un alias de **-p linux dnsdomainname**.

- Muestra la documentación del comando original:

```
tldr -p linux dnsdomainname
```



# gecho

Este comando es un alias de **-p linux echo**.

- Muestra la documentación del comando original:

```
tldr -p linux echo
```

# ged

Este comando es un alias de **-p linux ed**.

- Muestra la documentación del comando original:

```
tldr -p linux ed
```

# gegrep

Este comando es un alias de **-p linux egrep**.

- Muestra la documentación del comando original:

```
tldr -p linux egrep
```

# genv

Este comando es un alias de **-p linux env**.

- Muestra la documentación del comando original:

```
tldr -p linux env
```

# gexpand

Este comando es un alias de **-p linux expand**.

- Muestra la documentación del comando original:

```
tldr -p linux expand
```

# gexpr

Este comando es un alias de **-p linux expr**.

- Muestra la documentación del comando original:

```
tldr -p linux expr
```

# gfactor

Este comando es un alias de **-p linux factor**.

- Muestra la documentación del comando original:

```
tldr -p linux factor
```

# gfalse

Este comando es un alias de **-p linux false**.

- Muestra la documentación del comando original:

```
tldr -p linux false
```



# gfgrep

Este comando es un alias de **-p linux fgrep**.

- Muestra la documentación del comando original:

```
tldr -p linux fgrep
```

# gfind

Este comando es un alias de **-p linux find**.

- Muestra la documentación del comando original:

```
tldr -p linux find
```

# gfmt

Este comando es un alias de **-p linux fmt**.

- Muestra la documentación del comando original:

```
tldr -p linux fmt
```

# gfold

Este comando es un alias de **-p linux fold**.

- Muestra la documentación del comando original:

```
tldr -p linux fold
```

# gftp

Este comando es un alias de **-p linux ftp**.

- Muestra la documentación del comando original:

```
tldr -p linux ftp
```

# ggrep

Este comando es un alias de **-p linux grep**.

- Muestra la documentación del comando original:

```
tldr -p linux grep
```

# ggroups

Este comando es un alias de **-p linux groups**.

- Muestra la documentación del comando original:

```
tldr -p linux groups
```

# ghead

Este comando es un alias de **-p linux head**.

- Muestra la documentación del comando original:

```
tldr -p linux head
```



# ghostid

Este comando es un alias de **-p linux hostid**.

- Muestra la documentación del comando original:

```
tldr -p linux hostid
```

# ghostname

Este comando es un alias de **-p linux hostname**.

- Muestra la documentación del comando original:

```
tldr -p linux hostname
```

# gid

Este comando es un alias de **-p linux id**.

- Muestra la documentación del comando original:

```
tldr -p linux id
```

# gifconfig

Este comando es un alias de **-p linux ifconfig**.

- Muestra la documentación del comando original:

```
tldr -p linux ifconfig
```

# gindent

Este comando es un alias de **-p linux indent**.

- Muestra la documentación del comando original:

```
tldr -p linux indent
```

# ginstall

Este comando es un alias de **-p linux install**.

- Muestra la documentación del comando original:

```
tldr -p linux install
```

# gjoin

Este comando es un alias de **-p linux join**.

- Muestra la documentación del comando original:

```
tldr -p linux join
```

# gkill

Este comando es un alias de **-p linux kill**.

- Muestra la documentación del comando original:

```
tldr -p linux kill
```



# glibtool

Este comando es un alias de **-p linux libtool**.

- Muestra la documentación del comando original:

```
tldr -p linux libtool
```

# glibtoolize

Este comando es un alias de **-p linux libtoolize**.

- Muestra la documentación del comando original:

```
tldr -p linux libtoolize
```

# glink

Este comando es un alias de **-p linux link**.

- Muestra la documentación del comando original:

```
tldr -p linux link
```

# gln

Este comando es un alias de **-p linux ln**.

- Muestra la documentación del comando original:

```
tldr -p linux ln
```

# glocate

Este comando es un alias de **-p linux locate**.

- Muestra la documentación del comando original:

```
tldr -p linux locate
```

# glogger

Este comando es un alias de **-p linux logger**.

- Muestra la documentación del comando original:

```
tldr -p linux logger
```

# glogname

Este comando es un alias de **-p linux logname**.

- Muestra la documentación del comando original:

```
tldr -p linux logname
```

# gls

Este comando es un alias de **-p linux ls**.

- Muestra la documentación del comando original:

```
tldr -p linux ls
```



# gmake

Este comando es un alias de **-p linux make**.

- Muestra la documentación del comando original:

```
tldr -p linux make
```

# gmd5sum

Este comando es un alias de **-p linux md5sum**.

- Muestra la documentación del comando original:

```
tldr -p linux md5sum
```

# gmkdir

Este comando es un alias de **-p linux mkdir**.

- Muestra la documentación del comando original:

```
tldr -p linux mkdir
```

# gmkfifo

Este comando es un alias de **-p linux mkfifo**.

- Muestra la documentación del comando original:

```
tldr -p linux mkfifo
```

# gmknod

Este comando es un alias de **-p linux mknod**.

- Muestra la documentación del comando original:

```
tldr -p linux mknod
```

# gmkttemp

Este comando es un alias de **-p linux mktemp**.

- Muestra la documentación del comando original:

```
tldr -p linux mktemp
```

# gmv

Este comando es un alias de **-p linux mv**.

- Muestra la documentación del comando original:

```
tldr -p linux mv
```

# gnice

Este comando es un alias de **-p linux nice**.

- Muestra la documentación del comando original:

```
tldr -p linux nice
```



# gnl

Este comando es un alias de **-p linux nl**.

- Muestra la documentación del comando original:

```
tldr -p linux nl
```

# gnohup

Este comando es un alias de **-p linux nohup**.

- Muestra la documentación del comando original:

```
tldr -p linux nohup
```

# gnproc

Este comando es un alias de **-p linux nproc**.

- Muestra la documentación del comando original:

```
tldr -p linux nproc
```

# gnumfmt

Este comando es un alias de **-p linux numfmt**.

- Muestra la documentación del comando original:

```
tldr -p linux numfmt
```

# god

Este comando es un alias de **-p linux od**.

- Muestra la documentación del comando original:

```
tldr -p linux od
```

# gpaste

Este comando es un alias de **-p linux paste**.

- Muestra la documentación del comando original:

```
tldr -p linux paste
```

# gpathchk

Este comando es un alias de **-p linux pathchk**.

- Muestra la documentación del comando original:

```
tldr -p linux pathchk
```

# gping

Este comando es un alias de **-p linux ping**.

- Muestra la documentación del comando original:

```
tldr -p linux ping
```



# gping6

Este comando es un alias de **-p linux ping6**.

- Muestra la documentación del comando original:

```
tldr -p linux ping6
```

# gpinky

Este comando es un alias de **-p linux pinky**.

- Muestra la documentación del comando original:

```
tldr -p linux pinky
```

# gpr

Este comando es un alias de **-p linux pr**.

- Muestra la documentación del comando original:

```
tldr -p linux pr
```

# gprintenv

Este comando es un alias de **-p linux printenv**.

- Muestra la documentación del comando original:

```
tldr -p linux printenv
```

# gprintf

Este comando es un alias de **-p linux printf**.

- Muestra la documentación del comando original:

```
tldr -p linux printf
```

# gptx

Este comando es un alias de **-p linux ptx**.

- Muestra la documentación del comando original:

```
tldr -p linux ptx
```

# gpwd

Este comando es un alias de **-p linux pwd**.

- Muestra la documentación del comando original:

```
tldr -p linux pwd
```

# grcp

Este comando es un alias de **-p linux rcp**.

- Muestra la documentación del comando original:

```
tldr -p linux rcp
```



# greadlink

Este comando es un alias de **-p linux readlink**.

- Muestra la documentación del comando original:

```
tldr -p linux readlink
```

# grealpath

Este comando es un alias de **-p linux realpath**.

- Muestra la documentación del comando original:

```
tldr -p linux realpath
```

# grexec

Este comando es un alias de **-p linux rexec**.

- Muestra la documentación del comando original:

```
tldr -p linux rexec
```

# grlogin

Este comando es un alias de **-p linux rlogin**.

- Muestra la documentación del comando original:

```
tldr -p linux rlogin
```

# grm

Este comando es un alias de **-p linux rm**.

- Muestra la documentación del comando original:

```
tldr -p linux rm
```

# grmdir

Este comando es un alias de **-p linux rmdir**.

- Muestra la documentación del comando original:

```
tldr -p linux rmdir
```

# grsh

Este comando es un alias de **-p linux rsh**.

- Muestra la documentación del comando original:

```
tldr -p linux rsh
```

# gruncon

Este comando es un alias de **-p linux runcon**.

- Muestra la documentación del comando original:

```
tldr -p linux runcon
```



# gsed

Este comando es un alias de **-p linux sed**.

- Muestra la documentación del comando original:

```
tldr -p linux sed
```

# gseq

Este comando es un alias de **-p linux seq**.

- Muestra la documentación del comando original:

```
tldr -p linux seq
```

# gsha1sum

Este comando es un alias de **-p linux sha1sum**.

- Muestra la documentación del comando original:

```
tldr -p linux sha1sum
```

# gsha224sum

Este comando es un alias de **-p linux sha224sum**.

- Muestra la documentación del comando original:

```
tldr -p linux sha224sum
```

# gsha256sum

Este comando es un alias de **-p linux sha256sum**.

- Muestra la documentación del comando original:

```
tldr -p linux sha256sum
```

# gsha384sum

Este comando es un alias de **-p linux sha384sum**.

- Muestra la documentación del comando original:

```
tldr -p linux sha384sum
```

# gsha512sum

Este comando es un alias de **-p linux sha512sum**.

- Muestra la documentación del comando original:

```
tldr -p linux sha512sum
```

# gshred

Este comando es un alias de **-p linux shred**.

- Muestra la documentación del comando original:

```
tldr -p linux shred
```



# gshuf

Este comando es un alias de **-p linux shuf**.

- Muestra la documentación del comando original:

```
tldr -p linux shuf
```

# gsleep

Este comando es un alias de **-p linux sleep**.

- Muestra la documentación del comando original:

```
tldr -p linux sleep
```

# gsort

Este comando es un alias de **-p linux sort**.

- Muestra la documentación del comando original:

```
tldr -p linux sort
```

# gsplit

Este comando es un alias de **-p linux split**.

- Muestra la documentación del comando original:

```
tldr -p linux split
```

# gstat

Este comando es un alias de **-p linux stat**.

- Muestra la documentación del comando original:

```
tldr -p linux stat
```

# gstdbuf

Este comando es un alias de **-p linux stdbuf**.

- Muestra la documentación del comando original:

```
tldr -p linux stdbuf
```

# gstty

Este comando es un alias de **-p linux stty**.

- Muestra la documentación del comando original:

```
tldr -p linux stty
```

# gsum

Este comando es un alias de **-p linux sum**.

- Muestra la documentación del comando original:

```
tldr -p linux sum
```



# gsync

Este comando es un alias de **-p linux sync**.

- Muestra la documentación del comando original:

```
tldr -p linux sync
```

# gtac

Este comando es un alias de **-p linux tac**.

- Muestra la documentación del comando original:

```
tldr -p linux tac
```

# gtail

Este comando es un alias de **-p linux tail**.

- Muestra la documentación del comando original:

```
tldr -p linux tail
```

# gtalk

Este comando es un alias de **-p linux talk**.

- Muestra la documentación del comando original:

```
tldr -p linux talk
```

# gtar

Este comando es un alias de **-p linux tar**.

- Muestra la documentación del comando original:

```
tldr -p linux tar
```

# gtee

Este comando es un alias de **-p linux tee**.

- Muestra la documentación del comando original:

```
tldr -p linux tee
```

# gtelnet

Este comando es un alias de **-p linux telnet**.

- Muestra la documentación del comando original:

```
tldr -p linux telnet
```

# gtest

Este comando es un alias de **-p linux test**.

- Muestra la documentación del comando original:

```
tldr -p linux test
```



# gtftp

Este comando es un alias de **-p linux tftp**.

- Muestra la documentación del comando original:

```
tldr -p linux tftp
```

# gtime

Este comando es un alias de **-p linux time**.

- Muestra la documentación del comando original:

```
tldr -p linux time
```

# gtimeout

Este comando es un alias de **-p linux timeout**.

- Muestra la documentación del comando original:

```
tldr -p linux timeout
```

# gtouch

Este comando es un alias de **-p linux touch**.

- Muestra la documentación del comando original:

```
tldr -p linux touch
```

# gtr

Este comando es un alias de **-p linux tr**.

- Muestra la documentación del comando original:

```
tldr -p linux tr
```

# gtracroute

Este comando es un alias de **-p linux traceroute**.

- Muestra la documentación del comando original:

```
tldr -p linux traceroute
```

# gtrue

Este comando es un alias de **-p linux true**.

- Muestra la documentación del comando original:

```
tldr -p linux true
```

# gtruncate

Este comando es un alias de **-p linux truncate**.

- Muestra la documentación del comando original:

```
tldr -p linux truncate
```



# gtsort

Este comando es un alias de **-p linux tsort**.

- Muestra la documentación del comando original:

```
tldr -p linux tsort
```

# gtty

Este comando es un alias de **-p linux tty**.

- Muestra la documentación del comando original:

```
tldr -p linux tty
```

# guname

Este comando es un alias de **-p linux uname**.

- Muestra la documentación del comando original:

```
tldr -p linux uname
```

# gunexpand

Este comando es un alias de **-p linux unexpand**.

- Muestra la documentación del comando original:

```
tldr -p linux unexpand
```

# guniq

Este comando es un alias de **-p linux uniq**.

- Muestra la documentación del comando original:

```
tldr -p linux uniq
```

# gunits

Este comando es un alias de **-p linux units**.

- Muestra la documentación del comando original:

```
tldr -p linux units
```

# gunlink

Este comando es un alias de **-p linux unlink**.

- Muestra la documentación del comando original:

```
tldr -p linux unlink
```

# gupdatedb

Este comando es un alias de **-p linux updatedb**.

- Muestra la documentación del comando original:

```
tldr -p linux updatedb
```



# guptime

Este comando es un alias de **-p linux uptime**.

- Muestra la documentación del comando original:

```
tldr -p linux uptime
```

# gusers

Este comando es un alias de **-p linux users**.

- Muestra la documentación del comando original:

```
tldr -p linux users
```

# gvdir

Este comando es un alias de **-p linux vdir**.

- Muestra la documentación del comando original:

```
tldr -p linux vdir
```

# gwc

Este comando es un alias de **-p linux wc**.

- Muestra la documentación del comando original:

```
tldr -p linux wc
```

# gwhich

Este comando es un alias de **-p linux which**.

- Muestra la documentación del comando original:

```
tldr -p linux which
```

# gwho

Este comando es un alias de **-p linux who**.

- Muestra la documentación del comando original:

```
tldr -p linux who
```

# gwhoami

Este comando es un alias de **-p linux whoami**.

- Muestra la documentación del comando original:

```
tldr -p linux whoami
```

# gwhois

Este comando es un alias de **-p linux whois**.

- Muestra la documentación del comando original:

```
tldr -p linux whois
```



# gxargs

Este comando es un alias de **-p linux xargs**.

- Muestra la documentación del comando original:

```
tldr -p linux xargs
```

# gyes

Este comando es un alias de **-p linux yes**.

- Muestra la documentación del comando original:

```
tldr -p linux yes
```

# hidd

Daemon userland de la librería HID.

No debe invocarse manualmente.

Más información: <https://keith.github.io/xcode-man-pages/hidd.8.html>.

- Inicia el daemon:

```
hidd
```

# icalBuddy

Utilidad de línea de comandos para imprimir eventos y tareas desde la base de datos del calendario de macOS.

Más información: <https://hasseg.org/icalBuddy/>.

- Muestra los eventos de hoy más tarde:

```
icalBuddy --includeOnlyEventsFromNowOn eventsToday
```

- Muestra tareas no completadas:

```
icalBuddy uncompletedTasks
```

- Muestra una lista formateada y discriminada de acuerdo al calendario de todos los eventos en el día de hoy:

```
icalBuddy --formatOutput --separateByCalendar eventsToday
```

- Muestra las tareas para un número determinado de días:

```
icalBuddy --includeOnlyEventsFromNowOn "tasksDueBefore:today+{{days}}"
```

- Muestra los eventos en un rango de tiempo:

```
icalBuddy eventsFrom:{{start_date}} to:{{end_date}}
```

# InternetSharing

Configura Internet Sharing.

No debe invocarse manualmente.

Más información: <https://www.manpagez.com/man/8/InternetSharing/>.

- Inicia el daemon:

`InternetSharing`

# iostat

Informa las estadísticas de los dispositivos.

Más información: <https://ss64.com/mac/iostat.html>.

- Muestra estadísticas instantáneas de dispositivos (KB/t, transferencias y MB por segundo), estadísticas de CPU (porcentajes de tiempo empleado en modos usuario, sistema e inactivo) y promedios de carga del sistema (para los últimos 1, 5 y 15 min):

```
iostat
```

- Muestra solo estadísticas de dispositivos:

```
iostat -d
```

- Muestra informes incrementales de estadísticas de CPU y disco cada 2 segundos:

```
iostat 2
```

- Muestra las estadísticas del primer disco cada segundo de forma indefinida:

```
iostat -w 1 disk0
```

- Muestra las estadísticas del segundo disco cada 3 segundos, 10 veces:

```
iostat -w 3 -c 10 disk1
```

- Muestra utilizando la antigua muestra de `iostat`. Muestra los sectores transferidos por segundo, las transferencias por segundo, el promedio de milisegundos por transacción y las estadísticas de la CPU + los promedios de carga de la muestra por defecto:

```
iostat -o
```

- Muestra las estadísticas totales del dispositivo (KB/t: kilobytes por transferencia como antes, xfrs: número total de transferencias, MB: número total de megabytes transferidos):

```
iostat -I
```

# java\_home

Devuelve un valor para \$JAVA\_HOME o ejecuta un comando usando esta variable.

Más información: [https://www.unix.com/man-page/osx/1/java\\_home](https://www.unix.com/man-page/osx/1/java_home).

- Lista JVMs basadas en una versión específica:

```
java_home --version {{1.5+}}
```

- Lista JVMs basadas en una [arch]itectura específica:

```
java_home --arch {{i386}}
```

- Lista JVMs basadas en tareas específicas (por defecto CommandLine):

```
java_home --datamodel {{Applets|WebStart|BundledApp|JNI|
CommandLine}}
```

- Lista JVMs en formato XML:

```
java_home --xml
```

- Muestra la ayuda:

```
java_home --help
```

# lipo

Herramienta para el manejo de binarios universales Mach-O.

Más información: <https://keith.github.io/xcode-man-pages/lipo.1.html>.

- Crea un archivo universal a partir de dos archivos de una sola arquitectura:

```
lipo {{ruta/al/archivo/binario.x86_64}} {{ruta/al/archivo_binario.arm64e}} -create -output {{ruta/al/archivo_binario}}
```

- Lista todas las arquitecturas contenidas en un archivo universal:

```
lipo {{ruta/al/archivo_binario}} -archs
```

- Muestra información detallada sobre un archivo universal:

```
lipo {{ruta/al/archivo_binario}} -detailed_info
```

- Extrae un archivo de arquitectura única de un archivo universal:

```
lipo {{ruta/al/archivo_binario}} -thin {{arm64e}} -output {{ruta/al/archivo_binario.arm64e}}
```



# lldb

El depurador de bajo nivel LLVM.

Más información: <https://lldb.llvm.org/man/lldb.html>.

- Depura un ejecutable:

```
lldb "{{ejecutable}}"
```

- Adjunta lldb a un proceso en ejecución con un PID dado:

```
lldb -p {{pid}}
```

- Espera a que se inicie un nuevo proceso con un nombre determinado y adjuntarlo al mismo:

```
lldb -w -n "{{nombre_proceso}}"
```

# llvm-lipo

Este comando es un alias de **lipo**.

- Muestra la documentación del comando original:

```
tldr lipo
```

# look

Muestra las líneas que empiezan por un prefijo en un archivo ordenado.

Vea también: **grep**, **sort**.

Más información: <https://keith.github.io/xcode-man-pages/look.1.html>.

- Busca líneas que comiencen con un prefijo específico en un archivo específico:

```
look {{prefijo}} {{ruta/al/archivo}}
```

- Búsqueda insensible a mayúsculas y minúsculas solo en caracteres alfanuméricos:

```
look {{-f|--ignore-case}} {{-d|--alphanum}} {{prefijo}}  
{{ruta/al/archivo}}
```

- Especifica un carácter de terminación de cadena (espacio por defecto):

```
look {{-t|--terminate}} {{[,]}}
```

- Busca en /usr/share/dict/words (se asumen `--ignore-case` y `--alphanum`):

```
look {{prefijo}}
```

# lsappinfo

Controla y consulta CoreApplicationServices acerca del estado de la aplicación en el sistema.

Más información: <https://keith.github.io/xcode-man-pages/lsappinfo.8.html>.

- Lista todas las aplicaciones en ejecución con sus detalles:

```
lsappinfo list
```

- Muestra la aplicación frontal:

```
lsappinfo front
```

- Muestra la información de una aplicación específica:

```
lsappinfo info {{com.apple.calculator}}
```

# m

Navaja suiza para macOS.

Más información: <https://github.com/rgcr/m-cli>.

- Obtén el estado de la batería:

```
m battery status
```

- Desactiva Bluetooth:

```
m bluetooth off
```

- Lista los sistemas de archivos disponibles para formatear:

```
m disk filesystems
```

- Activa la función de ocultación automática del Dock:

```
m dock autohide YES
```

- Desactiva el cortafuegos:

```
m firewall disable
```

# machine

Imprime el tipo de máquina.

Más información: <https://keith.github.io/xcode-man-pages/machine.1.html>.

- Imprime la arquitectura de la CPU:

```
machine
```

# mas

Interfaz de línea de comandos para la Mac App Store.

Más información: <https://github.com/mas-cli/mas>.

- Inicia sesión en la Mac App Store por primera vez:

```
mas signin "{{usuario@ejemplo.com}}"
```

- Muestra todas las aplicaciones instaladas y sus identificadores de producto:

```
mas list
```

- Busca una aplicación, mostrando el precio junto a los resultados:

```
mas search " {{aplicación}}" --price
```

- Instala o actualiza una aplicación:

```
mas install {{identificador_producto}}
```

- Instala todas las actualizaciones pendientes:

```
mas upgrade
```

# md5

Calcula sumas de comprobación criptográficas MD5.

Más información: <https://keith.github.io/xcode-man-pages/md5.1.html>.

- Calcula la suma de comprobación MD5 de un archivo:

```
md5 {{ruta/al/archivo}}
```

- Calcula sumas de comprobación MD5 para varios archivos:

```
md5 {{ruta/al/archivo1 ruta/al/archivo2 ...}}
```

- Obtén sólo la suma de comprobación md5 (sin nombre de archivo):

```
md5 -q {{ruta/al/archivo}}
```

- Imprime una suma de comprobación de la cadena dada:

```
md5 -s "{{cadena}}"
```



# mdfind

Lista los archivos que coinciden con una consulta dada.

Más información: <https://keith.github.io/xcode-man-pages/mdfind.1.html>.

- Busca un archivo por su nombre:

```
mdfind -name {{archivo}}
```

- Busca un archivo por su contenido:

```
mdfind "{{consulta}}"
```

- Busca un archivo que contenga una cadena, en un directorio determinado:

```
mdfind -onlyin {{directorio}} "{{consulta}}"
```

# mist

MIST - macOS Installer Super Tool.

Descarga automáticamente Firmwares/Instaladores de macOS.

Más información: <https://github.com/ninxsoft/mist-cli>.

- Lista todos los firmwares de macOS disponibles para los Mac Silicon de Apple:

```
mist list firmware
```

- Lista todos los instaladores de macOS disponibles para Mac Intel, incluidos los instaladores universales para macOS Big Sur y versiones posteriores:

```
mist list installer
```

- Lista todos los instaladores de macOS compatibles con esta Mac, incluidos los instaladores universales de macOS Big Sur y posteriores:

```
mist list installer --compatible
```

- Lista todos los instaladores de macOS disponibles para Mac Intel, incluidas las betas tanto como también los instaladores universales para macOS Big Sur y versiones posteriores:

```
mist list installer --include-betas
```

- Lista solo el último instalador de macOS Sonoma para las Macs Intel, incluidos los instaladores universales de macOS Big Sur y posteriores:

```
mist list installer --latest "macOS Sonoma"
```

- Lista y exporta instaladores de macOS a un archivo CSV:

```
mist list installer --export "{{/ruta/a/  
archivo_con_datos_exportados.csv}}"
```

- Descarga el último firmware de macOS Sonoma para los Mac Silicon de Apple, con un nombre personalizado:

```
mist download firmware "macOS Sonoma" --firmware-name  
"{{Install %NAME% %VERSION%-%BUILD%.ipsw}}"
```

- Descarga una versión específica del instalador de macOS para Mac Intel, incluidos los instaladores universales de macOS Big Sur y posteriores:

```
mist download installer "{{13.5.2}}" application
```

# mktemp

Crea un archivo o directorio temporal.

Más información: <https://keith.github.io/xcode-man-pages/mktemp.1.html>.

- Crea un archivo temporal vacío e imprime su ruta absoluta:

```
mktemp
```

- Usa un directorio personalizado (por defecto la salida de `getconf DARWIN_USER_TEMP_DIR`, o `/tmp`):

```
mktemp --tmpdir={{/ruta/a/tempdir}}
```

- Usa una plantilla de ruta personalizada (las `X` se sustituyen por caracteres alfanuméricos aleatorios):

```
mktemp {{/tmp/ejemplo.XXXXXXXXXX}}
```

- Usa un prefijo de nombre de archivo personalizado:

```
mktemp -t {{ejemplo}}
```

- Crea un directorio temporal vacío e imprime su ruta absoluta:

```
mktemp --directory
```

# mysides

Añade, lista y elimina favoritos del Finder.

Más información: <https://github.com/mosen/mysides>.

- Lista favoritos de la barra lateral:

```
mysides list
```

- Añade un nuevo elemento al final de los favoritos de la barra lateral:

```
mysides add {{ejemplo}} {{file:///Usuarios/Compartidos/  
ejemplo}}
```

- Elimina un elemento por su nombre:

```
mysides remove {{ejemplo}}
```

- Añade el directorio actual a la barra lateral:

```
mysides add $(basename $(pwd)) file:///$(pwd)
```

- Elimina el directorio actual de la barra lateral:

```
mysides remove $(basename $(pwd))
```

# nettop

Muestra información actualizada sobre la red.

Más información: <https://keith.github.io/xcode-man-pages/nettop.1.html>.

- Monitoriza los sockets TCP y UDP de todas las interfaces:

```
nettop
```

- Monitoriza sockets TCP desde interfaces Loopback:

```
nettop -m {{tcp}} -t {{loopback}}
```

- Supervisa un proceso específico:

```
nettop -p "{{process_id|process_name}}"
```

- Muestra un resumen por proceso:

```
nettop -P
```

- Imprime 10 muestras de información de red:

```
nettop -l {{10}}
```

- Monitoriza los cambios cada 5 segundos:

```
nettop -d -s {{5}}
```

- Mientras se ejecuta nettop, lista los comandos interactivos:

```
h
```

- Muestra ayuda:

```
nettop -h
```

# nvrām

Manipula variables del firmware.

Más información: <https://keith.github.io/xcode-man-pages/nvrām.8.html>.

- Imprime todas las variables almacenadas en la NVRAM:

```
nvrām -p
```

- Imprime todas las variables almacenadas en la NVRAM usando el formato [x]ML:

```
nvrām -xp
```

- Modifica el valor de una variable del firmware:

```
sudo nvrām {{nombre}}="{{valor}}"
```

- Elimina una variable de firmware:

```
sudo nvrām -d {{nombre}}
```

- Clarifica todas las variables de firmware:

```
sudo nvrām -c
```

- Establece una variable de firmware de un [x]ML [f]ile específico:

```
sudo nvrām -xf {{ruta/al/archivo.xml}}
```

# open

Abre archivos, directorios y aplicaciones.

Más información: <https://keith.github.io/xcode-man-pages/open.1.html>.

- Abre un archivo con la aplicación asociada:

```
open {{archivo.ext}}
```

- Ejecuta una [a]plicación gráfica de macOS:

```
open -a "{{aplicacion}}"
```

- Ejecuta una aplicación gráfica de macOS basada en el identificador [b]undle (consulta `osascript` para obtenerlo fácilmente):

```
open -b {{com.domain.application}}
```

- Abre el directorio actual en Finder:

```
open .
```

- Abre un archivo en Finder:

```
open -R {{ruta/al/archivo}}
```

- Abre todos los archivos de una extensión determinada en el directorio actual con la aplicación asociada:

```
open {{*.ext}}
```

- Abre una [n]ueva instancia de una aplicación especificada mediante un identificador [b]undle:

```
open -n -b {{com.domain.application}}
```

# opensnoop

Herramienta que rastrea las aperturas de archivos en tu sistema.

Más información: <https://keith.github.io/xcode-man-pages/opensnoop.1m.html>.

- Imprime todos los archivos abiertos a medida que ocurren:

```
sudo opensnoop
```

- Rastrea todos los archivos abiertos por un proceso por su nombre:

```
sudo opensnoop -n "{{nombre_proceso}}"
```

- Rastrea todos los archivos abiertos por un proceso por PID:

```
sudo opensnoop -p {{PID}}
```

- Seguimiento de los procesos que abren un archivo especificado:

```
sudo opensnoop -f {{ruta/al/archivo}}
```



# osascript

Ejecuta AppleScript o JavaScript for Automation (JXA) desde la línea de comandos.

Más información: <https://keith.github.io/xcode-man-pages/osascript.1.html>.

- Ejecuta un comando AppleScript:

```
osascript -e "{{say 'Hello world'}}"
```

- Ejecuta varios comandos AppleScript:

```
osascript -e "{{say 'Hello'}}" -e "{{say 'world'}}"
```

- Ejecuta un archivo AppleScript compilado (\*.scpt), empaquetado (\*.scptd) o un archivo Applescript en texto plano (\*.applescript):

```
osascript {{ruta/al/apple.scpt}}
```

- Obtén el identificador del paquete de una aplicación (útil para open -b):

```
osascript -e 'id of app "{{Application}}"'
```

- Ejecuta un comando JavaScript:

```
osascript -l JavaScript -e "{{console.log('Hola mundo');}}"
```

- Ejecuta un archivo JavaScript:

```
osascript -l JavaScript {{ruta/al/script.js}}
```

# pbcopy

Copia datos de **stdin** al portapapeles.

Comparable a pulsar Cmd + C en el teclado.

Más información: <https://keith.github.io/xcode-man-pages/pbcopy.1.html>.

- Coloca el contenido de un archivo específico en el portapapeles:

```
pbcopy < {{ruta/al/archivo}}
```

- Coloca los resultados de un comando específico en el portapapeles:

```
find . -type t -name "*.png" | pbcopy
```

# pbpaste

Envía el contenido del portapapeles a la salida estándar.

Comparable a pulsar Cmd + V en el teclado.

Más información: <https://keith.github.io/xcode-man-pages/pbpaste.1.html>.

- Escribe el contenido del portapapeles en un archivo:

```
pbpaste > {{ruta/al/archivo}}
```

- Utiliza el contenido del portapapeles como entrada de un comando:

```
pbpaste | grep foo
```

# photoanalysisd

Analiza las bibliotecas de fotos para Memorias, Personas, y búsquedas basadas en escenas u objetos.

No debe invocarse manualmente.

Más información: <https://keith.github.io/xcode-man-pages/photoanalysisd.8.html>.

- Inicia el daemon:

```
photoanalysisd
```

# ping

Envía paquetes ICMP ECHO\_REQUEST a hosts de la red.

Más información: <https://keith.github.io/xcode-man-pages/ping.8.html>.

- Ping al host especificado:

```
ping "{{hostname}}"
```

- Ping a un host un número determinado de veces:

```
ping -c {{cuenta}} "{{host}}"
```

- Ping al host, especificando el intervalo en segundos entre peticiones (por defecto es 1 segundo):

```
ping -i {{segundos}} "{{host}}"
```

- Ping al host sin intentar buscar nombres simbólicos para las direcciones:

```
ping -n "{{host}}"
```

- Ping al host y hace sonar la campana cuando se recibe un paquete (si tu terminal lo soporta):

```
ping -a "{{host}}"
```

- Ping al host y muestra la hora en la que se ha recibido un paquete (esta opción es un añadido de Apple):

```
ping --apple-time "{{host}}"
```

# plutil

Ve, convierte, valida o edita archivos de listas de propiedades ("plist").

Más información: <https://keith.github.io/xcode-man-pages/plutil.1.html>.

- Muestra el contenido de uno o más archivos plist en un formato legible por humanos:

```
plutil -p {{archivo1.plist archivo2.plist ...}}
```

- Convierte uno o varios archivos plist a formato XML, sobrescribiendo los archivos originales:

```
plutil -convert xml1 {{archivo1.plist archivo2.plist ...}}
```

- Convierte uno o más archivos plist a formato binario, sobrescribiendo los archivos originales en su lugar:

```
plutil -convert binary1 {{archivo1.plist archivo2.plist ...}}
```

- Convierte un archivo plist a un formato diferente, escribiendo en un nuevo archivo:

```
plutil -convert {{xml1|binary1|json|swift|objc}} {{ruta/al/archivo.plist}} -o {{ruta/al/nuevo_archivo.plist}}
```

- Convierte un archivo plist a un formato diferente, escribiendo en stdout:

```
plutil -convert {{xml1|binary1|json|swift|objc}} {{ruta/al/archivo.plist}} -o -
```

# pod

Gestor de dependencias para proyectos Swift y Objective-C Cocoa.

Más información: <https://guides.cocoapods.org/terminal/commands.html>.

- Crea un Podfile para el proyecto actual con los contenidos por defecto:

```
pod init
```

- Descarga e instala todos los pods definidos en el Podfile (que no hayan sido instalados antes):

```
pod install
```

- Lista todos los pods disponibles:

```
pod list
```

- Muestra los pods obsoletos (de los instalados actualmente):

```
pod outdated
```

- Actualiza todos los pods instalados a su versión más reciente:

```
pod update
```

- Actualiza un pod específico (previamente instalado) a su versión más reciente:

```
pod update {{nombre_del_pod}}
```

- Elimina CocoaPods de un proyecto Xcode:

```
pod deintegrate {{proyecto_xcode}}
```

# port

Gestor de paquetes para macOS.

Más información: <https://www.macports.org>.

- Busca un paquete:

```
port search {{termino_de_busqueda}}
```

- Instala un paquete:

```
sudo port install {{nombre_de_paquete}}
```

- Lista los paquetes instalados:

```
port installed
```

- Actualiza port y trae la última lista de paquetes disponibles:

```
sudo port selfupdate
```

- Actualiza los paquetes desactualizados:

```
sudo port upgrade outdated
```

- Remueve versiones antiguas de paquetes instalados:

```
sudo port uninstall inactive
```



# ps

Información sobre los procesos en ejecución.

Más información: <https://keith.github.io/xcode-man-pages/ps.1.html>.

- Lista todos los procesos en ejecución:

```
ps aux
```

- Lista todos los procesos en ejecución incluyendo la cadena de comandos completa:

```
ps auxww
```

- Busca un proceso que coincida con una cadena:

```
ps aux | grep {{cadena}}
```

- Obtén el PID principal de un proceso:

```
ps -o ppid= -p {{pid}}
```

- Ordena los procesos por uso de memoria:

```
ps -m
```

- Ordena los procesos por uso de CPU:

```
ps -r
```

# rargs

Ejecuta un comando por cada línea de entrada estándar.

Similar a **xargs**, pero con soporte de coincidencia de patrones.

Más información: <https://github.com/lotabout/rargs>.

- Ejecuta un comando por cada línea de entrada, como **xargs** (`{0}` indica donde sustituir en el texto):

```
{{comando}} | rargs {{comando}} {0}
```

- Imprime los comandos que se ejecutarían en vez de ejecutarlos:

```
{{comando}} | rargs -e {{comando}} {0}
```

- Elimina la extensión `.bak` de todos los archivos de una lista:

```
{{comando}} | rargs -p '(.*)\.bak mv {0} {1}
```

- Ejecuta comandos en paralelo:

```
{{comando}} | rargs -w {{máxima_cantidad_de_procesos}}
```

- Lee líneas de entrada separadas por el caracter NUL (`\0`) en vez de LF (`\n`):

```
{{comando}} | rargs -0 {{comando}} {0}
```

# readlink

Sigue enlaces simbólicos y obtiene información sobre enlaces simbólicos.

Más información: <https://www.gnu.org/software/coreutils/readlink>.

- Imprime la ruta absoluta a la que apunta el enlace simbólico:

```
readlink {{ruta/al/archivo_symlink}}
```

# reboot

Reinicia el sistema.

Más información: <https://keith.github.io/xcode-man-pages/reboot.8.html>.

- Reinicia inmediatamente:

```
sudo reboot
```

- Reinicia inmediatamente sin apagar el sistema:

```
sudo reboot -q
```

# route

Manipula manualmente las tablas de enrutamiento.

Necesita ser root.

Más información: <https://keith.github.io/xcode-man-pages/route.8.html>.

- Añade una ruta a un destino a través de una puerta de enlace:

```
sudo route add "{{dirección_ip_destino}}"  
"{{dirección_puerta}}"
```

- Añade una ruta a una subred /24 a través de una puerta de enlace:

```
sudo route add "{{dirección_ip_subred}}/24"  
"{{dirección_puerta}}"
```

- Ejecuta en modo de prueba (no hace nada, sólo imprime):

```
sudo route -t add "{{dirección_ip_destino}}/24"  
"{{dirección_puerta}}"
```

- Elimina todas las rutas:

```
sudo route flush
```

- Elimina una ruta específica:

```
sudo route delete "{{dirección_ip_destino}}/24"
```

- Busca y muestra la ruta de un destino (nombre de host o dirección IP):

```
sudo route get "{{destino}}"
```

# SafeEjectGPU

Expulsa una GPU de forma segura.

Más información: <https://keith.github.io/xcode-man-pages/SafeEjectGPU.8.html>.

- Expulsa todas las GPUs:

```
SafeEjectGPU Eject
```

- Lista todas las GPUs conectadas:

```
SafeEjectGPU gpus
```

- Lista de aplicaciones que utilizan una GPU:

```
SafeEjectGPU gpuid {{GPU_ID}} apps
```

- Obtén el estado de una GPU:

```
SafeEjectGPU gpuid {{GPU_ID}} status
```

- Expulsa una GPU:

```
SafeEjectGPU gpuid {{GPU_ID}} Eject
```

- Inicia una aplicación en una GPU:

```
SafeEjectGPU gpuid {{GPU_ID}} LaunchOnGPU {{ruta/al/App.app}}
```

# say

Convierte texto a voz.

Más información: <https://keith.github.io/xcode-man-pages/say.1.html>.

- Di una frase en voz alta:

```
say "{{Me gusta andar en mi bicicleta.}}"
```

- Lee un archivo en voz alta:

```
say --input-file={{nombre_de_archivo.txt}}
```

- Di una frase con una voz y un ritmo de voz personalizados:

```
say --voice={{voz}} --rate={{palabras_por_minuto}} "{{Lo  
siento David, no puedo dejarte hacer eso.}}"
```

- Lista las voces disponibles (cada voz habla en un idioma distinto):

```
say --voice="?"
```

- Di algo en polaco:

```
say --voice={{Zosia}} "{{Litwo, ojczyzno moja!}}"
```

- Crea un archivo de audio con el texto hablado:

```
say --output-file={{nombre_de_fichero.aiff}} "{{Aquí están  
los locos.}}"
```

# scutil

Gestiona los parámetros de configuración del sistema.

Es necesario ser root para establecer la configuración.

Más información: <https://keith.github.io/xcode-man-pages/scutil.8.html>.

- Muestra la configuración DNS:

```
scutil --dns
```

- Muestra la configuración del proxy:

```
scutil --proxy
```

- Obtén el nombre del equipo:

```
scutil --get ComputerName
```

- Establece el nombre del equipo:

```
sudo scutil --set ComputerName {{nombre_ordenador}}
```

- Obtén el nombre del host:

```
scutil --get HostName
```

- Establece nombre del host:

```
scutil --set HostName {{nombre_host}}
```



# sed

Edita texto de manera programable.

Vea también: **awk**, **ed**.

Más información: <https://keith.github.io/xcode-man-pages/sed.1.html>.

- Reemplaza todas las veces que aparece **apple** (regex básico) por **mango** (regex básico) en todas las líneas de entrada e imprime el resultado en `stdout`:

```
{{command}} | sed 's/apple/mango/g'
```

- Ejecuta un script [f] específico e imprime el resultado en `stdout`:

```
{{command}} | sed -f {{ruta/al/archivo_script.sed}}
```

- Sustituye todas las apariciones de **manzana** (regex extendido) por **MANZANA** (regex extendido) en todas las líneas de entrada e imprime el resultado en `stdout`:

```
{{command}} | sed -E 's/(manzana)/\U\1/g'
```

- Imprime solo la primera línea en `stdout`:

```
{{command}} | sed -n '1p'
```

- Sustituye todas las apariciones de **manzana** (regex básico) por **mango** (regex básico) en un **fichero** y guarda una copia de seguridad del original en **fichero.bak**:

```
sed -i bak 's/manzana/mango/g' {{ruta/al/archivo}}
```

# shortcuts

Gestiona los accesos directos desde la línea de comandos en lugar de la aplicación **Shortcuts**.

Más información: <https://support.apple.com/guide/shortcuts-mac/run-shortcuts-from-the-command-line-apd455c82f02/mac>.

- Ejecuta el atajo especificado (Contar vacaciones):

```
shortcuts run "{{Contar vacaciones}}"
```

- Muestra todos los atajos:

```
shortcuts list
```

- Muestra todas las carpetas de accesos directos:

```
shortcuts list --folders
```

- Abre el acceso directo especificado (Contar vacaciones) en el editor Shortcuts:

```
shortcuts view "{{Contar vacaciones}}"
```

# shuf

Genera permutaciones aleatorias.

Más información: <https://keith.github.io/xcode-man-pages/shuf.1.html>.

- Ordena aleatoriamente las líneas de un fichero y muestra el resultado:

```
shuf {{nombre_archivo}}
```

- Sólo muestra las 5 primeras entradas del resultado:

```
shuf --head-count=5 {{nombre_archivo}}
```

- Escribe el resultado en otro archivo:

```
shuf {{nombre_archivo}} --output={{nombre_archivo_salida}}
```

- Genera números aleatorios en el rango 1-10:

```
shuf --input-range=1-10
```

# sips

Sistema de procesamiento de imágenes Apple Scriptable.

Imágenes Raster/Query y Perfiles ICC ColorSync.

Más información: <https://keith.github.io/xcode-man-pages/sips.1.html>.

- Especifica un directorio de salida para que los originales no se modifiquen:

```
sips --out {{ruta/al/directorio_salida}}
```

- Remuestrea la imagen al tamaño especificado, la relación de aspecto de la imagen puede verse alterada:

```
sips --resampleHeightWidth {{1920}} {{300}}  
{{archivo_imagen.ext}}
```

- Remuestrea la imagen para que la altura y la anchura no superen el tamaño especificado (fíjate en la Z mayúscula):

```
sips --resampleHeightWidthMax {{1920}} {{300}}  
{{archivo_imagen.ext}}
```

- Remuestrea todas las imágenes de un directorio para que se ajusten a una anchura de 960px (respetando la relación de aspecto):

```
sips --resampleWidth {{960}} {{ruta/al/imágenes}}
```

- Convierte una imagen de CMYK a RGB:

```
sips --matchTo "/System/Library/ColorSync/Profiles/Generic  
RGB Profile.icc" {{ruta/al/imagen.ext}} {{ruta/al/  
directorio_salida}}
```

- Elimina el perfil ICC ColorSync de una imagen:

```
sips --deleteProperty profile --  
deleteColorManagementProperties {{ruta/al/  
archivo_de_imagen.ext}}
```

# sntp

Un cliente para Simple Network Time Protocol.

Más información: <https://keith.github.io/xcode-man-pages/sntp.1>.

- Consulta un servidor SNTP especificado y muestra la hora:

```
sntp {{pool.ntp.org}}
```

- Sincroniza el reloj del sistema con un servidor SNTP especificado:

```
sudo sntp -S {{pool.ntp.org}}
```

- Activa el registro de depuración:

```
sntp -d {{pool.ntp.org}}
```

# spctl

Gestiona el subsistema de políticas de evaluación de seguridad.

Utilidad para gestionar Gatekeeper en macOS.

Más información: <https://keith.github.io/xcode-man-pages/spctl.8.html>.

- Desactiva Gatekeeper:

```
spctl --master-disable
```

- Añade una regla para permitir la ejecución de una aplicación (el etiquetado de la regla es opcional):

```
spctl --add --label {{nombre_regla}} {{ruta/al/archivo}}
```

- Activa Gatekeeper:

```
spctl --master-enable
```

- Lista todas las reglas del sistema:

```
spctl --list
```

# split

Divide un archivo en trozos.

Más información: <https://keith.github.io/xcode-man-pages/split.1.html>.

- Divide un archivo, cada división tiene 10 líneas (excepto la última división):

```
split -l 10 {{ruta/al/archivo}}
```

- Divide un fichero mediante una expresión regular. La línea que coincida será la primera línea del siguiente archivo de salida:

```
split -p {{cat|^[dh]og}} {{ruta/al/archivo}}
```

- Divide un archivo con 512 bytes en cada división (excepto en la última; utiliza 512k para kilobytes y 512m para megabytes):

```
split -b 512 {{ruta/al/archivo}}
```

- Divide un archivo en 5 archivos. El archivo se divide de forma que cada división tenga el mismo tamaño (excepto la última división):

```
split -n 5 {{ruta/al/archivo}}
```

# spotify

Una línea de comando para Spotify.

Más información: <https://github.com/hnarayanan/shpotify>.

- Encuentra una canción por su nombre y la reproduce:

```
spotify play {{song_name}}
```

- Encuentra una lista de reproducción por su nombre y la reproduce:

```
spotify play list {{playlist_name}}
```

- Pausa (o reanuda) la reproducción:

```
spotify pause
```

- Pasa a la siguiente canción de una lista de reproducción:

```
spotify next
```

- Cambia el volumen:

```
spotify vol {{up|down|value}}
```

- Muestra el estado de reproducción y los detalles de la canción:

```
spotify status
```



# sw\_vers

Imprime información sobre la versión del sistema operativo macOS.

Más información: [https://keith.github.io/xcode-man-pages/sw\\_vers.1.html](https://keith.github.io/xcode-man-pages/sw_vers.1.html).

- Imprime toda la información disponible (nombre del sistema operativo, número de versión y compilación):

```
sw_vers
```

- Imprime sólo el número de versión del sistema operativo:

```
sw_vers -productVersion
```

- Imprime sólo el identificador de compilación:

```
sw_vers -buildVersion
```

# systemsoundserverd

Daemon relacionado con Core Audio.

No debe invocarse manualmente.

- Inicia el daemon:

```
systemsoundserverd
```

# tail

Muestra la última parte de un archivo.

Vea también: **head**.

Más información: <https://keith.github.io/xcode-man-pages/tail.1.html>.

- Muestra las últimas líneas 'count' del archivo:

```
tail -n {{8}} {{ruta/al/archivo}}
```

- Imprime un archivo a partir de un número de línea específico:

```
tail -n +{{8}} {{ruta/al/archivo}}
```

- Imprime un recuento específico de bytes desde el final de un archivo determinado:

```
tail -c {{8}} {{ruta/al/archivo}}
```

- Imprime las últimas líneas de un archivo dado y sigue leyéndolo hasta Ctrl + C:

```
tail -f {{ruta/al/archivo}}
```

- Sigue leyendo el archivo hasta Ctrl + C, incluso si el archivo es inaccesible:

```
tail -F {{ruta/al/archivo}}
```

- Muestra las últimas líneas 'contadas' en 'archivo' y lo actualiza cada 'n' segundos:

```
tail -n {{8}} -s {{10}} -f {{ruta/al/archivo}}
```

# tart

Crea, ejecuta y gestiona máquinas virtuales (VM) macOS y Linux en Apple Silicon.

Más información: <https://github.com/cirruslabs/tart>.

- Extrae una imagen de VM remota:

```
tart pull {{acme.io/org/nombre:tag}}
```

- Clona una VM desde una fuente de imagen local o remota:

```
tart clone {{source-vm}} {{nombre-vm}}
```

- Crea una nueva Mac VM a partir de un archivo ipsw específico:

```
tart create --from-ipsw={{ultima|ruta/al/archivo.ipsw}}  
{{nombre-de-la-vm}}
```

- Ejecuta una máquina virtual existente:

```
tart run {{nombre-de-la-vm}}
```

- Ejecuta una máquina virtual existente con un directorio específico montado:

```
tart run --dir={{ruta/al/directorio}}:{{ruta/a/directorio  
local}} {{nombre-de-la-vm}}
```

- Lista máquinas virtuales:

```
tart list
```

- Obtén la dirección IP de una máquina virtual en ejecución:

```
tart ip {{nombre-de-la-vm}}
```

- Cambia la resolución de pantalla de una máquina virtual:

```
tart set {{nombre-de-la-vm}} --display {{640}}x{{400}}
```

# terminal-notifier

Envía notificaciones de usuario en macOS.

Más información: <https://github.com/julienXX/terminal-notifier>.

- Envía una notificación (sólo se necesita el mensaje):

```
terminal-notifier -group {{tldr-info}} -title {{TLDR}} -  
message '{{TLDR mola}}'
```

- Muestra datos transmitidos con un sonido:

```
echo '{{Datos de mensajes transmitidos!}}' | terminal-  
notifier -sound {{default}}
```

- Abre una URL al hacer clic en la notificación:

```
terminal-notifier -message '{{Comprueba tus acciones de  
Apple!}}' -open '{{http://finance.yahoo.com/q?s=AAPL}}'
```

- Abre una aplicación al hacer clic en la notificación:

```
terminal-notifier -message '{{Importados 42 contactos.}}' -  
activate {{com.apple.AddressBook}}
```

# textutil

Manipula archivos de texto en varios formatos.

Más información: <https://keith.github.io/xcode-man-pages/textutil.1.html>.

- Muestra información de `foo.rtf`:

```
textutil -info {{foo.rtf}}
```

- Convierte `foo.rtf` en `foo.html`:

```
textutil -convert {{html}} {{foo.rtf}}
```

- Convierte texto enriquecido a texto normal:

```
textutil {{foo.rtf}} -convert {{txt}}
```

- Convierte `foo.txt` en `foo.rtf`, usando la fuente Times con un tamaño 10:

```
textutil -convert {{rtf}} -font {{Times}} -fontsize {{10}}  
{{foo.txt}}
```

- Carga todos los archivos RTF en el directorio actual, concatena su contenido y escribe el resultado como `index.html` con el título HTML establecido en "Varios archivos":

```
textutil -cat {{html}} -title "Varios archivos" -output  
{{index.html}} *.rtf
```

# tmutil

Utilidad para gestionar las copias de seguridad de Time Machine. La mayoría de las acciones requieren privilegios de root.

Más información: <https://keith.github.io/xcode-man-pages/tmutil.8.html>.

- Establece una unidad HFS+ como destino de la copia de seguridad:

```
sudo tmutil setdestination {{ruta/al/punto_montaje_disco}}
```

- Establece un recurso compartido APF o SMB como destino de la copia de seguridad:

```
sudo tmutil setdestination "{{protocolo://  
usuario[:contraseña]@host/compartir}}"
```

- Añade el destino indicado a la lista de destinos:

```
sudo tmutil setdestination -a {{destino}}
```

- Activa copias de seguridad automáticas:

```
sudo tmutil enable
```

- Desactiva las copias de seguridad automáticas:

```
sudo tmutil disable
```

- Inicia una copia de seguridad, si ya no se está ejecutando, y libera el control del intérprete de comandos:

```
sudo tmutil startbackup
```

- Inicia una copia de seguridad y bloquearla hasta que termine:

```
sudo tmutil startbackup -b
```

- Detiene una copia de seguridad:

```
sudo tmutil stopbackup
```

# top

Muestra información dinámica en tiempo real sobre los procesos en ejecución.

Más información: <https://keith.github.io/xcode-man-pages/top.1.html>.

- Inicia top, todas las opciones están disponibles en la interfaz:

```
top
```

- Inicia top ordenando los procesos por tamaño de memoria interna (orden por defecto - ID del proceso):

```
top -o mem
```

- Inicia top ordenando los procesos primero por CPU, luego por tiempo de ejecución:

```
top -o cpu -0 time
```

- Inicia top mostrando sólo los procesos que pertenecen a un usuario determinado:

```
top -user {{nombre_usuario}}
```

- Muestra información de ayuda sobre comandos interactivos:

```
?
```



# uname

Imprime detalles sobre la máquina actual y el sistema operativo que se ejecuta en ella.

Nota: para obtener información adicional sobre el sistema operativo, pruebe el comando **sw\_vers**.

Más información: <https://keith.github.io/xcode-man-pages/uname.1.html>.

- Imprime el nombre del kernel:

```
uname
```

- Muestra la arquitectura del sistema y la información del procesador:

```
uname -mp
```

- Muestra el nombre, la versión y la versión del kernel:

```
uname -s rv
```

- Muestra el nombre de host del sistema:

```
uname -n
```

- Muestra toda la información disponible del sistema:

```
uname -a
```

# universalaccessd

Proporciona servicios de acceso universal.

No debe invocarse manualmente.

Más información: <https://keith.github.io/xcode-man-pages/universalaccessd.8.html>.

- Inicia el daemon:

```
universalaccessd
```

# uptime

Indica cuánto tiempo lleva funcionando el sistema y otras informaciones.

Más información: <https://keith.github.io/xcode-man-pages/uptime.1.html>.

- Imprime la hora actual, el tiempo de actividad, el número de usuarios conectados y otras informaciones:

```
uptime
```

# usernoted

Proporciona servicios de notificación.

No debe invocarse manualmente.

Más información: <https://keith.github.io/xcode-man-pages/usernoted.8.html>.

- Inicia el daemon:

```
usernoted
```

# uuidgen

Genera nuevas cadenas UUID (Identificador único universal).

Más información: <https://keith.github.io/xcode-man-pages/uuidgen.1.html>.

- Genera una cadena UUID:

```
uuidgen
```

# valet

Un entorno de desarrollo Laravel que permite alojar sitios a través de túneles locales en **http://<ejemplo>.test**.

Más información: <https://laravel.com/docs/valet>.

- Inicia el daemon valet:

```
valet start
```

- Registra el directorio de trabajo actual como ruta en la que Valet debe buscar sitios:

```
valet park
```

- Muestra las rutas 'aparcadas':

```
valet paths
```

- Sirve un único sitio en lugar de un directorio completo:

```
valet link {{nombre_aplicacion}}
```

- Comparte un proyecto a través de un túnel Ngrok:

```
valet share
```

# vm\_stat

Muestra estadísticas de memoria virtual.

Más información: [https://keith.github.io/xcode-man-pages/vm\\_stat.1.html](https://keith.github.io/xcode-man-pages/vm_stat.1.html).

- Muestra estadísticas de memoria virtual:

```
vm_stat
```

- Muestra informes cada 2 segundos durante 5 veces:

```
vm_stat -c {{5}} {{2}}
```

# vpnd

Escucha las conexiones VPN entrantes.

No debería ejecutar el programa manualmente.

Más información: <https://keith.github.io/xcode-man-pages/vpnd.8.html>.

- Inicia el daemon:

```
vpnd
```

- Ejecuta el daemon en primer plano:

```
vpnd -x
```

- Ejecuta el daemon en primer plano e imprime los registros en la terminal:

```
vpnd -d
```

- Ejecuta el daemon en primer plano, imprime los registros en la terminal y termina después de validar los argumentos:

```
vpnd -n
```

- Ejecuta el daemon para una configuración de servidor específica:

```
vpnd -i {{identificador_de_servidor}}
```

- Muestra ayuda:

```
vpnd -h
```



# W

Muestra quién está conectado y qué está haciendo.

Imprime el inicio de sesión del usuario, TTY, host remoto, tiempo de inicio de sesión, tiempo de inactividad, proceso actual.

Más información: <https://keith.github.io/xcode-man-pages/w.1.html>.

- Muestra información de los usuarios conectados:

```
w
```

- Muestra información de usuarios conectados sin encabezado:

```
w -h
```

- Muestra información sobre los usuarios conectados, ordenados por su tiempo de inactividad:

```
w -i
```

# wacaw

Herramienta de línea de comandos para macOS para capturar imágenes fijas y videos desde una cámara adjunta.

Más información: <https://webcam-tools.sourceforge.net>.

- Toma una foto desde la cámara web:

```
wacaw {{filename}}
```

- Graba un video:

```
wacaw --video {{filename}} --duration {{duration_in_seconds}}
```

- Toma una foto con resolución personalizada:

```
wacaw --width {{width}} --height {{height}} {{filename}}
```

- Copia imagen recién tomada al portapapeles:

```
wacaw --to-clipboard
```

- Lista de los dispositivos disponibles:

```
wacaw --list-devices
```

# warmd

Controla las cachés utilizadas durante el arranque y el inicio de sesión.

No debe invocarse manualmente.

Más información: <https://keith.github.io/xcode-man-pages/warmd.8.html>.

- Inicia el daemon:

```
warmd
```

# watchlistd

Gestiona la lista de seguimiento de la aplicación Apple TV.

No debe invocarse manualmente.

Más información: <https://keith.github.io/xcode-man-pages/watchlistd.8.html>.

- Inicia el daemon:

```
watchlistd
```

# WC

Cuenta líneas, palabras o bytes.

Más información: <https://keith.github.io/xcode-man-pages/wc.1.html>.

- Cuenta líneas en un archivo:

```
wc -l {{ruta/al/archivo}}
```

- Cuenta palabras en el archivo:

```
wc -w {{ruta/al/archivo}}
```

- Cuenta caracteres (bytes) en el archivo:

```
wc -c {{ruta/al/archivo}}
```

- Cuenta caracteres en el archivo (teniendo en cuenta los conjuntos de caracteres multibyte):

```
wc -m {{ruta/al/archivo}}
```

- Usa `stdin` para contar líneas, palabras y caracteres (bytes) en ese orden:

```
{{find .}} | wc
```

# webinspectord

Transmite comandos entre Web Inspector y objetos remotos como WKWebView.

No debe invocarse manualmente.

Más información: <https://keith.github.io/xcode-man-pages/webinspectord.8.html>.

- Inicia el daemon:

```
webinspectord
```

# whatis

Herramienta que busca palabras clave en un conjunto de archivos de base de datos que contienen descripciones breves de comandos del sistema.

Más información: <https://www.linfo.org/whatis.html>.

- Busca información sobre palabra clave:

```
whatis {{palabra_clave}}
```

- Busca información sobre varias palabras clave:

```
whatis {{palabra_clave1}} {{palabra_clave2}}
```

# whence

Un comando integrado de zsh para indicar cómo se interpretaría un comando dado.

Más información: <https://keith.github.io/xcode-man-pages/whence.1.html>.

- Interpreta `comando`, con expansión si se define como un `alias` (similar al `command -v` integrado):

```
whence "{{comando}}"
```

- Muestra tipo de `comando`, con localización si se define como una función, o binario (equivalente a los `type` y `command -V` integrados):

```
whence -v "{{comando}}"
```

- Igual que el anterior, excepto que muestra el contenido de las funciones del shell en lugar de la ubicación (equivalente al `which` integrado):

```
whence -c "{{comando}}"
```

- Igual que el anterior, pero muestra todas las apariciones en la ruta del comando (equivalente al `where` integrado):

```
whence -ca "{{comando}}"
```

- Busca un comando en la variable de entorno `PATH`, ignorando los comandos integrados, alias o funciones del shell (equivalente al comando `where`):

```
whence -p "{{comando}}"
```



# wifi-password

Obtiene la contraseña del Wi-Fi.

Más información: <https://github.com/rauchg/wifi-password>.

- Obtén la contraseña de la red Wi-Fi en la que está conectado actualmente:

```
wifi-password
```

- Obtén la contraseña para el Wi-Fi con un SSID específico:

```
wifi-password {{ssid}}
```

- Imprime solo la contraseña como salida:

```
wifi-password -q
```

# wifivelocityd

Asistente XPC para realizar acciones de contexto de sistema para el framework WiFiVelocity.

No debe invocarse manualmente.

Más información: <https://keith.github.io/xcode-man-pages/wifivelocityd.8.html>.

- Inicia el daemon:

```
wifivelocityd
```

# wps

Ayuda a AirPort a conectarse a una red mediante la Configuración inalámbrica protegida.

No debe invocarse manualmente.

Más información: <https://keith.github.io/xcode-man-pages/wps.8.html>.

- Inicia el daemon:

```
wps
```

# wwand

Daemon de configuración del dispositivo USB WWAN.

No debe invocarse manualmente.

Más información: <https://keith.github.io/xcode-man-pages/wwand.8.html>.

- Inicia el daemon:

```
wwand
```

# xartstorageremoted

El Daemon de Almacenamiento Remoto xART. Recibe las solicitudes de guardar/obtener del CoProcesador.

No debe ser invocado manualmente.

Más información: <https://keith.github.io/xcode-man-pages/xartstorageremoted.8.html>.

- Inicia el daemon:

```
xartstorageremoted
```

# xattr

Utilidad para trabajar con atributos extendidos del sistema de ficheros.

Más información: <https://keith.github.io/xcode-man-pages/xattr.1.html>.

- Lista atributos extendidos clave:valor para un archivo dado:

```
xattr -l {{archivo}}
```

- Escribe un atributo para un archivo determinado:

```
xattr -w {{atributo_clave}} {{atributo_valor}} {{archivo}}
```

- Elimina un atributo de un archivo determinado:

```
xattr -d {{com.apple.quarantine}} {{archivo}}
```

- Elimina todos los atributos extendidos de un archivo determinado:

```
xattr -c {{archivo}}
```

- Elimina recursivamente un atributo en un directorio determinado:

```
xattr -rd {{clave_atributo}} {{directorio}}
```

# xcode-select

Cambia entre diferentes versiones de Xcode y las herramientas incluidas para desarrolladores.

También se utiliza para actualizar la ruta a Xcode si se mueve después de la instalación.

Más información: <https://developer.apple.com/library/archive/technotes/tn2339/index.html>.

- Instala las herramientas de línea de comandos de Xcode:

```
xcode-select --install
```

- Selecciona una ruta determinada como directorio de desarrollador activo:

```
xcode-select --switch {{ruta/al/Xcode.app/Contents/  
Developer}}
```

- Selecciona una instancia de Xcode determinada y utiliza su directorio de desarrollador como directorio activo:

```
xcode-select --switch {{ruta/al/archivo/Xcode_file.app}}
```

- Muestra el directorio de desarrollador seleccionado:

```
xcode-select --print-path
```

- Descarta cualquier directorio de desarrolladores especificado por el usuario para que se encuentre mediante el mecanismo de búsqueda predeterminado:

```
sudo xcode-select --reset
```

# xcodebuild

Construye proyectos Xcode.

Más información: <https://developer.apple.com/library/archive/technotes/tn2339/index.html>.

- Construye espacio de trabajo:

```
xcodebuild -workspace  
{{nombre_del_espacio_de_trabajo.workspace}} -scheme  
{{nombre_scheme}} -configuration {{nombre_configuration}}  
clean build SYMROOT={{ruta_SYMROOT}}
```

- Construye proyecto:

```
xcodebuild -target {{nombre_target}} -configuration  
{{nombre_configuration}} clean build SYMROOT={{ruta_SYMROOT}}
```

- Muestra los SDKs:

```
xcodebuild -showsdk
```



# xcodes runtimes

Gestiona los tiempos de ejecución de Xcode Simulator.

Más información: <https://github.com/xcodesorg/xcodes>.

- Muestra todos los tiempos de ejecución del simulador disponibles:

```
xcodes runtimes --include-betas
```

- Descarga un simulador de un tiempo de ejecución:

```
xcodes runtimes download {{nombre_del_tiempo_de_ejecución}}
```

- Descarga e instala un simulador de un tiempo de ejecución:

```
xcodes runtimes install {{nombre_del_tiempo_de_ejecución}}
```

# xcrun

Ejecuta o localiza herramientas de desarrollo y propiedades.

Más información: <https://keith.github.io/xcode-man-pages/xcrun.1.html>.

- Localiza y ejecuta una herramienta desde el directorio activo de desarrolladores:

```
xcrun {{herramienta}} {{argumentos}}
```

- Muestra resultados detallados:

```
xcrun {{herramienta}} {{argumentos}} --verbose
```

- Busca una herramienta para un SDK:

```
xcrun --sdk {{nombre_de_sdk}}
```

- Busca una herramienta para una cadena de herramientas:

```
xcrun --toolchain {{nombre_de_cadena}}
```

- Muestra ayuda:

```
xcrun --help
```

- Muestra la versión:

```
xcrun --version
```

# xctool

Herramienta para construir proyectos Xcode.

Más información: <https://github.com/facebookarchive/xctool>.

- Construye un proyecto sin ningún espacio de trabajo:

```
xctool -project {{TuProyecto.xcodeproj}} -scheme  
{{SuEsquema}} build
```

- Construye un proyecto que forma parte de un espacio de trabajo:

```
xctool -workspace {{TuEspacioDeTrabajo.xcworkspace}} -scheme  
{{TuEsquema}} build
```

- Limpia, construye y ejecuta todas las pruebas:

```
xctool -workspace {{TuEspacioTrabajo.xcworkspace}} -scheme  
{{TuEsquema}} clean build test
```

# xed

Abre archivos para editarlos en Xcode.

Más información: <https://keith.github.io/xcode-man-pages/xed.1.html>.

- Abre archivo en Xcode:

```
xed {{archivo1}}
```

- Abre archivo(s) en Xcode, lo crea si no existe:

```
xed --create {{nombre_archivo1}}
```

- Abre un archivo en Xcode y salta a la línea número 75:

```
xed --line 75 {{nombre_archivo}}
```

# xip

Crea o expande archivos comprimidos en un archivo xip seguro.

Sólo los archivos firmados por Apple son de confianza, por lo que esta herramienta no debe utilizarse para crear archivos comprimidos.

Más información: <https://keith.github.io/xcode-man-pages/xip.1.html>.

- Expande el archivo en el directorio de trabajo actual:

```
xip --expand {{ruta/al/archivo.xip}}
```

# xml2man

Compila MPGL a mdoc.

Más información: <https://developer.apple.com/library/archive/documentation/DeveloperTools/Conceptual/HeaderDoc/mpgl/mpgl.html>.

- Compila un archivo MPGL a una página man visible:

```
xml2man {{ruta/al/archivo_de_comando.mxml}}
```

- Compila un archivo MPGL a un archivo de salida específico:

```
xml2man {{ruta/al/archivo_servicio.mxml}} {{ruta/al/archivo_servicio.7}}
```

- Compila un archivo MPGL a un archivo de salida específico, sobrescribiéndolo si ya existe:

```
xml2man -f {{ruta/al/archivo_funcion.mxml}} {{ruta/al/archivo_de_funcion.3}}
```

# xsand

Daemon de gestión del sistema de archivos Xsan. Proporciona servicios para el sistema de archivos Xsan.

No debe invocarse manualmente.

Más información: <https://developer.apple.com/support/downloads/Xsan-Management-Guide.pdf>.

- Inicia el daemon:

```
xsand
```

# xsltproc

Transforma XML con XSLT para producir una salida (normalmente HTML o XML).

Más información: <http://www.xmlsoft.org/xslt/xsltproc.html>.

- Transforma un archivo XML con una hoja de estilos XSLT específica:

```
xsltproc --output {{ruta/al/archivo_salida.html}} {{ruta/al/
archivo_hoja_estilo.xslt}} {{ruta/al/archivo.xml}}
```

- Pasa un valor a un parámetro de la hoja de estilos:

```
xsltproc --output {{ruta/al/archivo_salida.html}} --
stringparam "{{nombre}}" "{{value}}" {{ruta/al/
archivo_hoja_estilo.xslt}} {{ruta/al/archivo_xml.xml}}
```



# yaa

Crea y manipula archivos YAA.

Más información: <https://keith.github.io/xcode-man-pages/yaa.1.html>.

- Crea un archivo a partir de un directorio:

```
yaa archive -d {{ruta/al/directorio}} -o {{ruta/al/archivo_de_salida.yaa}}
```

- Crea un archivo a partir de un fichero:

```
yaa archive -i {{ruta/al/archivo}} -o {{ruta/al/archivo_de_salida.yaa}}
```

- Extrae un archivo al directorio actual:

```
yaa extract -i {{ruta/al/archivo.yaa}}
```

- Lista el contenido de un archivo:

```
yaa list -i {{ruta/al/archivo.yaa}}
```

- Crea un archivo con un algoritmo de compresión específico:

```
yaa archive -a {{algorithm}} -d {{ruta/al/directorio}} -o {{ruta/al/archivo_de_salida.yaa}}
```

- Crea un archivo con un tamaño de bloque de 8 MB:

```
yaa archive -b 8m -d {{ruta/al/directorio}} -o {{ruta/al/archivo_de_salida.yaa}}
```

# yabai

Un administrador de ventanas en mosaico para macOS basado en la partición de espacio binario.

Más información: <https://github.com/koekeishiya/yabai/wiki>.

- Establece la disposición a bsp:

```
yabai -m config layout {{bsp}}
```

- Establece el espacio de la ventana en 10pt:

```
yabai -m config window_gap {{10}}
```

- Habilita opacidad:

```
yabai -m config window_opacity on
```

- Deshabilita la sombra de la ventana:

```
yabai -m config window_shadow off
```

- Habilita la barra de estado:

```
yabai -m config status_bar on
```

Windows

# cd

Muestra el directorio actual o cambia a un directorio.

Más información: <https://learn.microsoft.com/windows-server/administration/windows-commands/cd>.

- Muestra la ruta (path) del directorio actual:

```
cd
```

- Cambia a la raíz de la unidad actual:

```
cd \
```

- Cambia al directorio superior:

```
cd ..
```

- Cambia a un directorio específico en el mismo disco:

```
cd {{ruta\al\directorio}}
```

- Cambia a un directorio específico en otro disco:

```
cd /d {{C}}:{{ruta\al\directorio}}
```

# choco

El gestor de paquetes Chocolatey.

Algunos subcomandos como **install** tienen su propia documentación de uso.

Más información: <https://chocolatey.org>.

- Ejecuta un comando Chocolatey:

```
choco {{comando}}
```

- Llama a la ayuda general:

```
choco -?
```

- Llama a la ayuda sobre un comando específico:

```
choco {{comando}} -?
```

- Revisa la versión de Chocolatey:

```
choco --version
```

# chrome

Este comando es un alias de **chromium**.

Más información: <https://chrome.google.com>.

- Muestra la documentación del comando original:

```
tldr chromium
```

# cinst

Este comando es un alias de **choco install**.

Más información: <https://docs.chocolatey.org/en-us/choco/commands/install>.

- Muestra la documentación del comando original:

```
tldr choco install
```

# clear

En PowerShell, este comando es un alias de **Clear-Host**.

- Muestra la documentación del comando original:

```
tldr clear-host
```



# clist

Este comando es un alias de **choco list**.

Más información: <https://docs.chocolatey.org/en-us/choco/commands/list>.

- Muestra la documentación del comando original:

```
tldr choco list
```

# cls

Borra la pantalla.

Más información: <https://learn.microsoft.com/windows-server/administration/windows-commands/cls>.

- Borra la pantalla:

```
cls
```

# cmd

El intérprete de comandos de Windows.

Más información: <https://learn.microsoft.com/windows-server/administration/windows-commands/cmd>.

- Inicia una sesión shell interactiva:

```
cmd
```

- Ejecuta [c]omandos específicos:

```
cmd /c {{echo Hola Mundo}}
```

- Ejecuta un script específico:

```
cmd {{ruta\al\script.bat}}
```

- Ejecuta comandos específicos y luego entrar en un shell interactivo:

```
cmd /k {{echo Hola Mundo}}
```

- Inicia una sesión shell interactiva donde `echo` está desactivado en la salida de comandos:

```
cmd /q
```

- Inicia una sesión de shell interactiva con la expansión [v]ariable retardada activada o desactivada:

```
cmd /v:{{on|off}}
```

- Inicia una sesión shell interactiva con [e]xtensiones de comando activadas o desactivadas:

```
cmd /e:{{on|off}}
```

- Inicia una sesión shell interactiva with con la codificación [u]nicode utilizada:

```
cmd /u
```

# cpush

Este comando es un alias de **choco push**.

Más información: <https://docs.chocolatey.org/en-us/create/commands/push>.

- Muestra la documentación del comando original:

```
tldr choco-push
```

# cuninst

Este comando es un alias de **choco uninstall**.

Más información: <https://docs.chocolatey.org/en-us/choco/commands/uninstall>.

- Muestra la documentación del comando original:

```
tldr choco uninstall
```

# del

Elimina uno o más archivos.

En PowerShell, este comando es un alias de **Remove-Item**. Esta documentación se basa en la versión del símbolo del sistema (**cmd**) de **del**.

Más información: <https://learn.microsoft.com/windows-server/administration/windows-commands/del>.

- Muestra la documentación del comando PowerShell equivalente:

```
tldr remove-item
```

- Elimina uno o más archivos o patrones separados por espacios:

```
del {{patrón_del_archivo}}
```

- Solicita confirmación antes de borrar cada archivo:

```
del {{patrón_del_archivo}} /p
```

- Fuerza la eliminación de archivos de sólo lectura:

```
del {{patrón_del_archivo}} /f
```

- Elimina recursivamente archivos de todos los subdirectorios:

```
del {{patrón_del_archivo}} /s
```

- Elimina archivos que coincidan con un comodín sin confirmación:

```
del {{patrón_del_archivo}} /q
```

- Muestra la ayuda y la lista de atributos disponibles:

```
del /?
```

- Elimina archivos en función de los atributos especificados:

```
del {{patrón_del_archivo}} /a {{atributo}}
```

# dir

Lista el contenido del directorio.

Más información: <https://learn.microsoft.com/windows-server/administration/windows-commands/dir>.

- Muestra el contenido del directorio actual:

```
dir
```

- Muestra el contenido de un directorio determinado:

```
dir {{ruta\al\directorio}}
```

- Muestra el contenido del directorio actual, incluidos los ocultos:

```
dir /a
```

- Muestra el contenido de un directorio determinado, incluidos los ocultos:

```
dir {{ruta\al\directorio}} /a
```

- Muestra una lista de directorios y archivos, sin información adicional:

```
dir /b
```

# exit

Sale de la instancia CMD actual o del archivo por lotes actual.

Más información: <https://learn.microsoft.com/windows-server/administration/windows-commands/exit>.

- Sale de la instancia CMD actual:

```
exit
```

- Sale del conjunto de instrucciones del archivo por lotes actual:

```
exit /b
```

- Sale usando un código de salida específico:

```
exit {{2}}
```



# getmac

Muestra las direcciones MAC de un sistema.

Más información: <https://learn.microsoft.com/windows-server/administration/windows-commands/getmac>.

- Muestra las direcciones MAC del sistema actual:

```
getmac
```

- Muestra los detalles en un formato específico:

```
getmac /fo {{table|list|csv}}
```

- Excluye la cabecera en la lista de salida:

```
getmac /nh
```

- Muestra las direcciones MAC de un equipo remoto:

```
getmac /s {{nombre_host}} /u {{nombredeusuario}} /p {{contraseña}}
```

- Muestra las direcciones MAC con información detallada:

```
getmac /v
```

- Muestra información de uso detallada:

```
getmac /?
```

# iwr

Este comando es un alias de **invoke-webrequest**.

Más información: <https://learn.microsoft.com/powershell/module/microsoft.powershell.utility/invoke-webrequest>.

- Muestra la documentación del comando original:

```
tldr invoke-webrequest
```

# mkdir

Crea un directorio.

Más información: <https://learn.microsoft.com/windows-server/administration/windows-commands/mkdir>.

- Crea un directorio:

```
mkdir {{ruta\al\directorio}}
```

- Crea un árbol de directorios anidado de forma recursiva:

```
mkdir {{ruta\al\sub_directorio}}
```

# nvm

Instala, desinstala o cambia entre versiones de Node.js.

Admite números de versión como "12.8" o "v16.13.1", y etiquetas como "stable", "system", etc.

Más información: <https://github.com/coreybutler/nvm-windows>.

- Instala una versión específica de Node.js:

```
nvm install {{versión_de_node}}
```

- Establece la versión por defecto de Node.js (debe ejecutarse como Administrador):

```
nvm use {{versión_de_node}}
```

- Lista todas las versiones disponibles de Node.js y destaca la versión por defecto:

```
nvm list
```

- Lista todas las versiones remotas de Node.js:

```
nvm ls-remote
```

- Desinstalación de una versión determinada de Node.js:

```
nvm uninstall {{versión_de_node}}
```

# ospp.vbs

Instala, activa y administra versiones con licencia por volumen de productos Microsoft Office.

Nota: este comando puede anular, desactivar y/o eliminar tu volumen actual de versiones de productos Office con licencia, así que procede con cautela.

Más información: <https://learn.microsoft.com/deployoffice/vlactivation/tools-to-manage-volume-activation-of-office>.

- Instala una clave de producto (Nota: sustituye a la clave existente):

```
cscript ospp.vbs /inpkey:{{clave_producto}}
```

- Desinstala una clave de producto instalada con los cinco últimos dígitos de la clave de producto:

```
cscript ospp.vbs /unpkey:{{clave_producto}}
```

- Establece un nombre de host KMS:

```
cscript ospp.vbs /sethst:{{ip|nombre_host}}
```

- Establece un puerto KMS:

```
cscript ospp.vbs /setprt:{{puerto}}
```

- Activa las claves de producto de Office instaladas:

```
cscript ospp.vbs /act
```

- Muestra la información de licencia de las claves de producto instaladas:

```
cscript ospp.vbs /dstatus
```

# pwsh where

Este comando es un alias de **Where-Object**.

Más información: <https://learn.microsoft.com/powershell/module/microsoft.powershell.core/where-object>.

- Muestra la documentación del comando original:

```
tldr Where-Object
```

# rd

Este comando es un alias de **rmdir**.

Más información: <https://learn.microsoft.com/windows-server/administration/windows-commands/rd>.

- Muestra la documentación del comando original:

```
tldr rmdir
```

# sls

Este comando es un alias de **Select-String**.

Más información: <https://learn.microsoft.com/powershell/module/microsoft.powershell.utility/select-string>.

- Muestra la documentación del comando original:

```
tldr select-string
```



# Test-NetConnection

Muestra información de diagnóstico de una conexión.

Este comando solo se puede utilizar a través de PowerShell.

Más información: <https://learn.microsoft.com/powershell/module/nettcpip/test-netconnection>.

- Prueba una conexión y muestra resultados detallados:

```
Test-NetConnection -InformationLevel Detailed
```

- Prueba una conexión a un host remoto con un número de puerto específico:

```
Test-NetConnection -ComputerName {{ip_o_nombre_del_host}} -  
Port {{número_de_puerto}}
```

