

# tldr pages book

Simplified and community-driven man pages

*Generated on Mon Nov 20 18:39:39 2023*

Website: <https://tldr.sh>

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

# Android

# am

Gerenciador de atividades do Android (Activity Manager).

Mais informações: <https://developer.android.com/studio/command-line/adb#am>.

- Inicia uma activity específica:

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

- Inicia uma activity e passa dados para ela:

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

- Inicia uma activity correspondente a uma ação e categoria específicas:

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

- Converte uma intent em uma URI:

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

# bugreport

Mostra um relatório de bugs do Android.

Esse comando só pode ser utilizado por meio de **adb shell**.

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

- Mostra um relatório completo de bugs de um dispositivo Android:

```
bugreport
```

# bugreportz

Gera um relatório de bugs do Android em formato .zip.

Esse comando só pode ser utilizado por meio de **adb shell**.

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

- Mostra um relatório completo de bugs de um dispositivo Android em formato .zip:

```
bugreportz
```

- Mostra o progresso de bugreportz em execução:

```
bugreportz -p
```

- Mostra a versão de bugreportz:

```
bugreportz -v
```

- Mostra a ajuda:

```
bugreportz -h
```

# cmd

Gerenciador de serviços do Android (service manager).

Mais informações: [https://cs.android.com/android/platform/superproject/+main:frameworks/native/cmds/cmd/](https://cs.android.com/android/platform/superproject/+/main:frameworks/native/cmds/cmd/).

- Lista todos os serviços em execução:

```
cmd -l
```

- Chama um serviço específico:

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

- Chama um serviço com parâmetros:

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

# dalvikvm

A máquina virtual Java do Android.

Mais informações: <https://source.android.com/devices/tech/dalvik>.

- Inicia um programa Java:

```
dalvikvm -classpath {{caminho/para/arquivo.jar}}  
{{nome_da_classe}}
```

# dumpsys

Fornece informações sobre os serviços do sistema Android.

Este comando só pode ser usado através de **adb shell**.

Mais informações: <https://developer.android.com/studio/command-line/dumpsys>.

- Gera um diagnóstico de todos os serviços do sistema:

```
dumpsys
```

- Gera um diagnóstico de um serviço do sistema específico:

```
dumpsys {{servico}}
```

- Lista todos os serviços que o **dumpsys** pode obter informações:

```
dumpsys -l
```

- Lista argumentos específicos-de-um-serviço para um serviço:

```
dumpsys {{servico}} -h
```

- Omite um serviço em específico do diagnóstico:

```
dumpsys --skip {{servico}}
```

- Especifica um periodo de timeout (por padrão é 10s):

```
dumpsys -t {{segundos}}
```



# getprop

Obtém informações sobre propriedades do sistema Android (system props).

Mais informações: <https://manned.org/getprop>.

- Mostra todas as propriedades do sistema:

```
getprop
```

- Mostra o valor de uma propriedade específica:

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

- Mostra o nível de API:

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

- Mostra a versão do Android:

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

- Mostra o modelo do dispositivo:

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

- Mostra o status de desbloqueio OEM:

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

- Mostra o endereço MAC da placa de Wi-Fi do dispositivo:

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

# input

Envia códigos de evento ou gestos de toque para um dispositivo Android.

Esse comando só pode ser usado através de **adb shell**.

Mais informações: <https://developer.android.com/reference/android/view/KeyEvent.html#constants> 1.

- Envia um código de evento de um caractere para um dispositivo Android:

```
input keyevent {{codigo_de_evento}}
```

- Envia texto para um dispositivo Android (%s representa espaços):

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

- Envia um único toque para um dispositivo Android:

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

- Envia um gesto de deslizar para um dispositivo Android:

```
input swipe {{x_inicio}} {{y_inicio}} {{x_fim}} {{y_fim}}  
{{duração_em_ms}}
```

- Envia um pressionamento longo usando gesto de deslizar para um dispositivo Android:

```
input swipe {{x_pos}} {{y_pos}} {{x_pos}} {{y_pos}}  
{{duração_em_ms}}
```

# logcat

Despeja um registro de mensagens do sistema.

Mais informações: <https://developer.android.com/studio/command-line/logcat>.

- Exibe a saída do registro:

```
logcat
```

- Salva a saída da mensagem de registro em um arquivo:

```
logcat -f {{caminho/para/arquivo}}
```

- Exibe apenas linhas em que a mensagem de registro corresponda a uma expressão regular:

```
logcat --regex {{expressao_regular}}
```

- Exibe logs para um PID específico:

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

- Exibe logs de processo de um pacote específico:

```
logcat --pid=$(pidof -s {{pacote}})
```

# pkg

Gerenciador de pacotes para o Termux.

Mais informações: [https://wiki.termux.com/wiki/Package\\_Management](https://wiki.termux.com/wiki/Package_Management).

- Atualiza todos os pacotes instalados:

```
pkg upgrade
```

- Instala um pacote:

```
pkg install {{pacote}}
```

- Desinstala um pacote:

```
pkg uninstall {{pacote}}
```

- Reinstala um pacote:

```
pkg reinstall {{pacote}}
```

- Busca por um pacote:

```
pkg search {{pacote}}
```

# pm

Executa ações e consultas em pacotes de apps instalados no dispositivo.

Mais informações: <https://developer.android.com/studio/command-line/adb#pm>.

- Exibe uma lista com todos os apps instalados:

```
pm list packages
```

- Exibe uma lista com todos os apps do sistema instalado:

```
pm list packages -s
```

- Exibe uma lista com todos os apps de terceiros instalados:

```
pm list packages -3
```

- Exibe uma lista com todos os apps cujos nomes estejam incluídos em palavras-chave:

```
pm list packages {{palavras_chave}}
```

- Exibe o caminho para o APK de um app:

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

# screencap

Tira uma screenshot do display mobile.

Este comando apenas pode ser usado através de **adb shell**.

Mais informações: <https://developer.android.com/studio/command-line/adb#screencap>.

- Tira uma screenshot:

```
screencap {{caminho/para/arquivo}}
```

# settings

Exibe, edita e apaga configurações do sistema Android.

Mais informações: <https://adbinstaller.com/commands/adb-shell-settings-5b670d5ee7958178a2955536>.

- Exibe a lista de configurações no namespace `global`:

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

- Obtém o valor de uma configuração específica:

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

- Edita o valor de uma configuração:

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

- Apaga uma configuração:

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

# wm

Exibe informações da tela de um dispositivo Android.

Esse comando só pode ser usado através de **adb shell**.

Mais informações: <https://adbinstaller.com/commands/adb-shell-wm-5b672b17e7958178a2955538>.

- Mostra o tamanho da tela de um dispositivo Android:

```
wm {{size}}
```

- Mostra a densidade de pixels da tela de um dispositivo Android:

```
wm {{density}}
```



Common

# Exclamation mark

Mecanismo interno do bash para substituir por um comando existente no histórico.

Mais informações: <https://www.gnu.org/software/bash/manual/bash.html#Event-Designators>.

- Substitui com o comando anterior e execute com o sudo:

```
sudo !!
```

- Substitui com o comando baseado no número da linha do `history`:

```
!{{número}}
```

- Substitui com o comando que foi usado algumas linhas do histórico atrás:

```
!-{{número}}
```

- Substitui com o comando mais recente que começa com o texto:

```
!{{texto}}
```

- Substitui com os argumentos do último comando:

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

# 2to3

Conversão automática de código Python 2 para Python 3.

Mais informações: <https://docs.python.org/3/library/2to3.html>.

- Mostrar as alterações que seriam feitas sem fazê-las de fato (simulação):

```
2to3 {{caminho/para/arquivo.py}}
```

- Converter um arquivo feito em Python 2 para Python 3:

```
2to3 --write {{caminho/para/arquivo.py}}
```

- Converter recurso específico de Python 2 para Python 3:

```
2to3 --write {{caminho/para/arquivo.py}} --fix={{raw_input}}  
--fix={{print}}
```

- Converter todos os recursos de Python 2 para Python 3, exceto as que especificadas:

```
2to3 --write {{caminho/para/arquivo.py}} --nofix={{has_key}}  
--nofix={{isinstance}}
```

- Mostrar a lista de todas os recursos disponíveis que podem ser convertidas de Python 2 para Python 3:

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

- Converter todos os arquivos feitos em Python 2 em um diretório para Python 3:

```
2to3 --output-dir={{caminho/para/arquivos_python3}} --write-  
unchanged-files --nobackups {{caminho/para/arquivos_python2}}
```

- Executar 2to3 com múltiplas threads:

```
2to3 --processes={{4}} --output-dir={{caminho/para/  
arquivos_python3}} --write --nobackups --no-diff {{caminho/  
para/arquivos_python2}}
```

# 7z

Um compactador de arquivos com alta taxa de compressão.

Mais informações: <https://manned.org/7z>.

- Compacta um arquivo ou diretório:

```
7z a {{arquivo_compactado.7z}} {{caminho/
arquivo_ou_diretório}}
```

- Criptografa um arquivo existente (incluindo cabeçalhos):

```
7z a {{arquivo_criptografado.7z}} -p{{senha}} -mhe=on
{{arquivo.7z}}
```

- Descompacta um arquivo mantendo a estrutura de diretórios original:

```
7z x {{arquivo.7z}}
```

- Descompacta um arquivo em um diretório especificado pelo usuário:

```
7z x {{caminho/para/arquivo.7z}} -o{{caminho/para/diretório}}
```

- Descompacta um arquivo para a saída padrão:

```
7z x {{caminho/para/arquivo_compactado.7z}} -so
```

- Compacta utilizando um tipo específico de arquivamento/compressão:

```
7z a -t{{zip|gzip|bzip2|tar}} {{caminho/para/
arquivo_compactado.7z}} {{caminho/para/arquivo_ou_diretório}}
```

- Exibe o conteúdo de um arquivo:

```
7z l {{arquivo.7z}}
```

- Exibe os tipos de arquivamento/compressão disponíveis:

```
7z i
```

# 7za

Um compactador de arquivos com alta taxa de compressão.

Versão compacta do **7z**, com suporte para menos tipos de arquivamento/compressão.

Mais informações: <https://manned.org/7za>.

- Compactar um arquivo ou diretório:

```
7za a {{caminho/para/arquivo_compactado.7z}} {{caminho/para/arquivo_ou_diretório}}
```

- Criptografa um arquivo existente (incluindo cabeçalhos):

```
7za a {{caminho/para/arquivo_criptografado.7z}} -p{{senha}} -mhe=on {{caminho/para/arquivo_compactado.7z}}
```

- Descompactar um arquivo mantendo a estrutura de diretórios original:

```
7za x {{caminho/para/arquivo_compactado.7z}}
```

- Descompacta um arquivo em um diretório especificado pelo usuário:

```
7za x {{caminho/para/arquivo_compactado.7z}} -o{{caminho/para/diretório}}
```

- Descompacta um arquivo para a saída padrão:

```
7za x {{caminho/para/arquivo_compactado.7z}} -so
```

- Compacta utilizando um tipo específico de arquivamento/compressão:

```
7za a -t{{zip|gzip|bzip2|tar}} {{caminho/para/arquivo_compactado.7z}} {{caminho/arquivo_ou_diretório}}
```

- Exibe o conteúdo de um arquivo:

```
7za l {{caminho/para/arquivo_compactado.7z}}
```

- Exibe os tipos de arquivamento/compressão disponíveis:

```
7za i
```

# 7zr

Um compactador de arquivos com alta taxa de compressão.

Versão do **7z** com suporte apenas para o formato **.7z**.

Mais informações: <https://manned.org/7zr>.

- Compactar um arquivo ou diretório:

```
7zr a {{caminho/para/arquivo_compactado.7z}} {{caminho/para/arquivo_ou_diretorio}}
```

- Criptografa um arquivo existente (incluindo cabeçalhos):

```
7zr a {{arquivo_criptografado.7z}} -p{{senha}} -mhe={{on}} {{caminho/para/arquivo_compactado.7z}}
```

- Descompacta um arquivo mantendo a estrutura de diretórios original:

```
7zr x {{arquivo_compactado.7z}}
```

- Descompacta um arquivo em um diretório especificado pelo usuário:

```
7zr x {{caminho/para/arquivo_compactado.7z}} -o{{caminho/para/diretorio}}
```

- Descompacta um arquivo para a saída padrão:

```
7zr x {{caminho/para/arquivo_compactado.7z}} -so
```

- Exibe o conteúdo de um arquivo:

```
7zr l {{caminho/para/arquivo_compactado.7z}}
```

- Exibe os tipos de arquivamento/compressão disponíveis:

```
7zr i
```

[

Avalia condição.

Retorna 0 se a condição for verdadeira, 1 se for falsa.

Mais informações: <https://www.gnu.org/software/bash/manual/bash.html#index-test>.

- Testa se uma determinada variável é igual a uma determinada string:

```
[ "{{variavel}}" == "{{/bin/zsh}}" ]
```

- Testa se uma determinada variável é igual/diferente/maior que/menor que/maior ou igual/menor ou igual a um determinado inteiro:

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

- Testa se uma determinada variável tem um valor [n]ão vazio:

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

- Testa se uma determinada variável é vazia:

```
[ -z "{{variavel}}" ]
```

- Testa se um arquivo existe:

```
[ -f "{{caminho/para/arquivo}}" ]
```

- Testa se um [d]iretório existe:

```
[ ! -d "{{caminho/para/diretorio}}" ]
```

- Testa se um determinado arquivo ou diretório [e]xiste:

```
[ -e {{caminho/para/arquivo_ou_diretorio}} ]
```

# [[

Verifica tipos de arquivos e compara valores.

Retorna 0 se a condição é verdadeira, 1 se a condição é falsa.

Mais informações: [https://www.gnu.org/software/bash/manual/bash.html#index-\\_005b\\_005b](https://www.gnu.org/software/bash/manual/bash.html#index-_005b_005b).

- Testa se uma determinada variável é igual/diferente a uma determinada string:

```
[[ ${variável} {==|!=} "{{string}}" ]]
```

- Testa se uma determinada string está em conformidade com um determinado padrão/regex:

```
[[ ${variável} {==|~=} {{padrão}} ]]
```

- Testa se uma determinada variável é igual [eq]/não igual [ne]/maior que [gt]/menor que [lt]/maior ou igual que [ge]/menor ou igual que [le] um determinado número:

```
[[ ${variável} -{eq|ne|gt|lt|ge|le} {{número}} ]]
```

- Testa se uma determinada variável tem um valor [n]ão vazio:

```
[[ -n ${variável} ]]
```

- Testa se determinada variável tem um valor va[z]io:

```
[[ -z ${variável} ]]
```

- Testa se um determinado arquivo existe[f]:

```
[[ -f {{caminho/para/arquivo}} ]]
```

- Testa se um determinado [d]iretório existe:

```
[[ -d {{caminho/para/diretório}} ]]
```

- Testa se um determinado arquivo ou diretório [e]xiste:

```
[[ -e {{caminho/para/arquivo_ou_diretório}} ]]
```



# aapt

Ferramenta Android de empacotamento de recursos.

Compila e empacota recursos de um aplicativo Android.

Mais informações: [https://elinux.org/Android\\_aapt](https://elinux.org/Android_aapt).

- Lista os arquivos contidos em um arquivo APK:

```
aapt list {{caminho/para/aplicativo.apk}}
```

- Exibe os metadados de um aplicativo (versão, permissão, etc.):

```
aapt dump badging {{caminho/para/aplicativo.apk}}
```

- Cria um novo arquivo APK com os arquivos especificados no diretório:

```
aapt package -F {{caminho/para/aplicativo.apk}} {{caminho/para/diretório}}
```

# ab

Ferramenta da Apache para realizar benchmarking e testes de carga em servidores web.

Mais informações: <https://httpd.apache.org/docs/current/programs/ab.html>.

- Executar 100 requisições HTTP do tipo GET para uma determinada URL:

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

- Executar 100 requisições HTTP do tipo GET para uma determinada URL, executando 10 requisições simultâneas de cada vez:

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

- Utilizar a funcionalidade HTTP Keep Alive, permitindo que várias requisições sejam feitas em uma sessão HTTP:

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

- Definir o tempo total do benchmarking, em segundos:

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

# abduco

Gerenciador de sessão no terminal.

Mais informações: <http://www.brain-dump.org/projects/abduco/>.

- Lista sessões:

```
abduco
```

- Anexa à uma sessão, criando-a se não existir:

```
abduco -A {{nome}} {{bash}}
```

- Anexa à uma sessão com `dvtm`, criando-a se não existir:

```
abduco -A {{nome}}
```

- Desanexa de uma sessão:

```
Ctrl + \
```

- Anexa à uma sessão no modo read-only (somente leitura):

```
abduco -Ar {{nome}}
```

# ac

Imprime estatísticas do tempo usuários conectados.

Mais informações: <https://man.openbsd.org/ac>.

- Imprime a quantidade de tempo que o usuário atual está conectado em horas:

```
ac
```

- Imprime a quantidade de tempo que os usuários estão conectado em horas:

```
ac -p
```

- Imprime a quantidade de tempo que um usuário específico está conectado em horas:

```
ac -p {{nome_do_usuario}}
```

- Imprime a quantidade de tempo que um usuário específico está conectado em horas por dia (com total):

```
ac -dp {{nome_do_usuario}}
```

# ack

Uma ferramenta de pesquisa similar ao grep, otimizada para programadores.

Mais informações: <https://beyondgrep.com/documentation>.

- Procurar por arquivos que contenham o termo "foo":

```
ack {{foo}}
```

- Procurar por arquivos em uma linguagem específica:

```
ack --ruby {{each_with_object}}
```

- Contar o número total de correspondências para o termo "foo":

```
ack -ch {{foo}}
```

- Mostrar o nome dos arquivos contendo o termo "foo" e o número de correspondências em cada arquivo:

```
ack -cl {{foo}}
```

# acme.sh

Script shell script que implementa o protocolo cliente ACME, uma alternativa para o certbot.

Veja também **acme.sh dns**.

Mais informações: <https://github.com/acmesh-official/acme.sh>.

- Emite um certificado usando o modo webroot:

```
acme.sh --issue --domain {{exemplo.com}} --webroot {{/caminho/para/webroot}}
```

- Emite um certificado para múltiplos domínios usando o modo standalone na porta 80:

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

- Emite um certificado usando o modo standalone TLS na porta 443:

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

- Emite um certificado usando uma configuração válida Nginx:

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

- Emite um certificado usando uma configuração válida Apache:

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

- Emite um certificado wildcard (\*) usando o modo DNS\_API automático:

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

- Instala os arquivos dos certificados em um local específico (útil para renovação automática do certificado):

```
acme.sh --install-cert -d {{exemplo.com}} --key-file {{/caminho/para/exemplo.com.key}} --fullchain-file {{/caminho/para/exemplo.com.cer}} --reloadcmd {{"systemctl force-reload nginx"}}
```

# act

Executa GitHub Actions localmente utilizando Docker.

Mais informações: <https://github.com/nektos/act>.

- Lista acoes disponiveis:

```
act -l
```

- Executa evento padrão:

```
act
```

- Executa evento especifico:

```
act {{tipo_de_evento}}
```

- Executa acao especifica:

```
act -a {{acao_id}}
```

- Nao executa realmente as acoes (i.e. um dry run):

```
act -n
```

- Mostra verbose logs:

```
act -v
```

# adb install

Android Debug Bridge Install: Instalar apps em uma instância do Android emulator ou dispositivos conectados.

Mais informações: <https://developer.android.com/studio/command-line/adb>.

- Instala um app Android em um emulador/dispositivo:

```
adb install {{caminho/para/arquivo.apk}}
```

- Reinstala um app existente, mantendo seus dados:

```
adb install -r {{caminho/para/arquivo.apk}}
```

- Concede todas as permissões listadas no manifesto do app:

```
adb install -g {{caminho/para/arquivo.apk}}
```

- Atualiza rapidamente um app já instalado atualizando apenas as partes do APK que mudaram:

```
adb install --fastdeploy {{caminho/para/arquivo.apk}}
```



# adb shell

Android Debug Bridge Shell: Executar remotamente comandos shell em instâncias do emulador Android ou dispositivos Android conectados.

Mais informações: <https://developer.android.com/studio/command-line/adb>.

- Inicia um shell interativo remoto no emulador/dispositivo:

```
adb shell
```

- Obtém todas as propriedades do emulador ou dispositivo:

```
adb shell getprop
```

- Reverte todas as permissões de tempo de execução para o padrão:

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

- Revoga uma permissão perigosa para um aplicação:

```
adb shell pm revoke {{pacote}} {{permissao}}
```

- Aciona um evento:

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

- Limpa os dados da aplicação no emulador/dispositivo:

```
adb shell pm clear {{pacote}}
```

- Inicia uma atividade no emulador/dispositivo:

```
adb shell am start -n {{pacote}}/{{atividade}}
```

- Inicia atividade "home" no emulador/dispositivo:

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

# adb

Android Debug Bridge: Comunica com uma instância do emulador Android emulador ou dispositivos conectados.

Alguns subcomandos tais como **adb shell** possuem sua própria documentação de uso.

Mais informações: <https://developer.android.com/studio/command-line/adb>.

- Checa se o servidor adb está em execução e o iniciar:

```
adb start-server
```

- Encerra o processo do servidor adb:

```
adb kill-server
```

- Inicia uma shell remota no emulador/dispositivo desejado:

```
adb shell
```

- Instala um app Android no emulador/dispositivo:

```
adb install -r {{caminho/para/arquivo.apk}}
```

- Copia um arquivo/pasta do dispositivo desejado:

```
adb pull {{caminho/para/arquivo_ou_pasta_no_dispositivo}}  
{{caminho/para/pasta_de_destino_local}}
```

- Copia um arquivo/pasta para o dispositivo desejado:

```
adb push {{caminho/para/arquivo_ou_pasta_local}} {{caminho/  
para/pasta_de_destino_no_dispositivo}}
```

- Exibe a lista de dispositivos conectados:

```
adb devices
```

# afconvert

Converte entre os formatos de arquivo AFF e Raw.

Mais informações: <https://manned.org/afconvert.1>.

- Usa uma extensão específica (padrão: aff):

```
afconvert -a {{extensão}} {{caminho/para/arquivo_de_entrada}}  
{{caminho/para/arquivo_de_saida1 caminho/para/  
arquivo_de_saida2 ...}}
```

- Usa um nível específico de compressão (padrão: 7):

```
afconvert -X{{0..7}} {{caminho/para/arquivo_de_entrada}}  
{{caminho/para/arquivo_de_saida1 caminho/para/  
arquivo_de_saida2 ...}}
```

# ag

The Silver Searcher. Parecido com o ack, mas com um foco em ser ainda mais rápido.

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

- Acha arquivos que contém "foo" e imprime as linhas correspondentes no contexto:

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

- Acha arquivos que contém "foo" em um diretório específico:

```
ag {{foo}} {{caminho/para/arquivo}}
```

- Acha arquivos que contém "foo", mas lista somente os nomes dos arquivos:

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

- Acha arquivos que contém "FOO" sem diferença de caixa e imprime somente o correspondente em vez de a linha inteira:

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

- Acha "foo" em arquivos cujo nome corresponde a "bar":

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

- Acha arquivos cujo conteúdo corresponde à expressão regular:

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

- Acha arquivos cujo nome corresponde a "foo":

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

# agate

Um simples servidor para o protocolo de internet Gemini.

Mais informações: <https://github.com/mbrubeck/agate>.

- Executa e gera uma chave privada e um certificado:

```
agate --content {{caminho/para/conteudo/}} --addr {[::]:  
1965}} --addr {{0.0.0.0:1965}} --hostname {{example.com}} --  
lang {{en-US}}
```

- Executa o servidor:

```
agate {{caminho/para/arquivo}}
```

- Exibe opções de ajuda:

```
agate -h
```

# age-keygen

Gera par de chaves **age**.

Veja **age** para como criptografar/descriptografar arquivos.

Mais informações: <https://manned.org/age-keygen>.

- Gera um par de chaves, salva em um arquivo não criptografado e imprime a chave pública para `stdout`:

```
age-keygen --output {{caminho/para/arquivo}}
```

- Converte uma identidade para um destinatário e imprime a chave pública para `stdout`:

```
age-keygen -y {{caminho/para/arquivo}}
```

# age

Uma simples, moderna e segura ferramenta de criptografia de arquivos.

Mais informações: <https://github.com/FiloSottile/age>.

- Gera um arquivo criptografado que pode ser descriptografado com uma frase-chave:

```
age --passphrase --output {{caminho/para/
arquivo_criptografado}} {{caminho/para/
arquivo_descriptografado}}
```

- Gera um par de chaves, salvando a chave privada em um arquivo não criptografado e imprimindo a chave pública para stdout:

```
age-keygen --output {{caminho/para/arquivo}}
```

- Criptografa arquivo com uma ou mais chaves públicas que são inseridas como literais:

```
age --recipient {{chave_pública_1}} --recipient
{{chave_pública_2}} {{caminho/para/arquivo_descriptografado}}
--output {{caminho/para/arquivo_criptografado}}
```

- Criptografa arquivo com uma ou mais chaves públicas que são especificadas no arquivo do destinatário:

```
age --recipients-file {{caminho/para/arquivo_destinatário}}
{{caminho/para/arquivo_descriptografado}} --output {{caminho/
para/arquivo_criptografado}}
```

- Descriptografa um arquivo com uma frase-chave:

```
age --decrypt --output {{caminho/para/
arquivo_descriptografado}} {{caminho/para/
arquivo_criptografado}}
```

- Descriptografa um arquivo com um arquivo chave privada:

```
age --decrypt --identity {{caminho/para/
arquivo_chave_privada}} --output {{caminho/para/
arquivo_descriptografado}} {{caminho/para/
arquivo_criptografado}}
```

# airmon-ng

Ativa modo de monitoramento em dispositivos de rede sem-fio.

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

- Lista os dispositivos sem-fio e seus respectivos estados:

```
sudo airmon-ng
```

- Liga o modo de monitoramento para um dispositivo específico:

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

- Encerra processos problemáticos que usam dispositivos sem-fio:

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

- Desativa o modo de monitoramento para um dispositivo específico:

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



# airpaste

Compartilhar mensagens e arquivos na mesma rede.

Mais informações: <https://github.com/mafintosh/airpaste>.

- Esperar por mensagens e mostrá-las quando recebidas:

```
airpaste
```

- Enviar texto:

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

- Enviar arquivo:

```
airpaste < {{caminho/para/arquivo}}
```

- Receber arquivo:

```
airpaste > {{caminho/para/arquivo}}
```

- Criar/Entrar em canal:

```
airpaste {{nome_do_canal}}
```

# alacritty

Terminal multiplataforma, acelerado por GPU.

Mais informações: <https://github.com/alacritty/alacritty>.

- Abre uma nova janela do Alacritty:

```
alacritty
```

- Roda em um diretório específico:

```
alacritty --working-directory {{caminho/para/diretório}}
```

- Roda um comando em uma nova janela do Alacritty:

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

- Especifica um arquivo de configuração alternativo (\$XDG\_CONFIG\_HOME/alacritty/alacritty.yml por padrão):

```
alacritty --config-file {{caminho/para/config.yml}}
```

- Executa com configuração ao vivo habilitada (pode também ser habilitada por padrão no alacritty.yml):

```
alacritty --live-config-reload --config-file {{caminho/para/config.yml}}
```

# alex

Uma ferramenta que captura escrita insensível e sem consideração.

Ajuda a encontrar no texto, frases favorecedoras de gênero, polarizantes, relacionadas à raça, insensíveis à religião e outras frases desiguais.

Mais informações: <https://github.com/get-alex/alex>.

- Analisa o texto do `stdin`:

```
echo {{A rede dele parece boa}} | alex --stdin
```

- Analisa todos arquivos no diretório atual:

```
alex
```

- Analisa um arquivo específico:

```
alex {{arquivo_do_texto.md}}
```

- Analisa todos arquivos em Markdown exceto `example.md`:

```
alex *.md !{{example.md}}
```

# alias

Cria apelidos -- palavras que são substituídas por um comando.

Apelidos expiram ao final da sessão atual do shell de comando, a menos que sejam definidos no arquivo de configuração do shell, por exemplo `~/.bashrc`.

Mais informações: <https://tldp.org/LDP/abs/html/aliases.html>.

- Criar um apelido:

```
alias {{apelido}}="{{comando}}"
```

- Visualizar o comando associado a um determinado apelido:

```
alias {{apelido}}
```

- Remover um apelido:

```
unalias {{apelido}}
```

- Exibir todos os apelidos definidos:

```
alias -p
```

- Tornar o comando `rm` interativo:

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

- Criar o apelido `la` como um atalho para `ls -a`:

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

# amass

Ferramenta de Attack Surface Mapping (mapeamento de superfície de ataque) e Asset Discovery (descoberta de asset) em profundidade.

Alguns subcomandos como **amass db** tem sua propria documentacao de uso.

Mais informações: <https://github.com/OWASP/Amass>.

- Executa um subcomando Amass:

```
amass {{subcomando}}
```

- Mostra ajuda geral:

```
amass -help
```

- Mostra ajuda de um subcomando Amass (como `intel`, `enum`, etc.):

```
amass -help {{subcomando}}
```

- Mostra a versão:

```
amass -version
```

# ansible

Gerencia grupos de computadores remotamente utilizando SSH. (use o arquivo `/etc/ansible/hosts` para adicionar novos grupos/hosts).

Alguns subcomando como **ansible galaxy** possuem a própria documentação de uso.

Mais informações: <https://www.ansible.com/>.

- Lista os hosts pertencentes a um grupo:

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

- Realiza o ping de um grupo de hosts invocando o módulo ping:

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

- Exibe fatos sobre um grupo de hosts invocando o módulo setup:

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

- Executa um comando em um grupo de hosts invocando o módulo command com argumentos:

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

- Executa um comando com privilégios administrativos:

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

- Executa um comando usando um arquivos de inventário customizado:

```
ansible {{grupo}} -i {{arquivo_inventario}} -m command -a  
'{{meu_comando}}'
```

- Lista os grupos presentes em um inventário:

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

# apg

Criar senhas aleatórias arbitrariamente complexas.

Mais informações: <https://manned.org/apg>.

- Criar senha aleatória (tamanho padrão para as senhas é 8 caracteres):

```
apg
```

- Criar senha com pelo menos 1 símbolo (S), 1 número (N), 1 letra maiúscula (C), 1 letra minúscula (L):

```
apg -M SNCL
```

- Criar uma senha com 16 caracteres:

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

- Criar senha com tamanho máximo de 16 caracteres:

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

- Criar uma senha que não aparece em um dicionário provido pelo usuário:

```
apg -r {{caminho/para/arquivo_de_dicionario}}
```

# apm

Gerenciador de pacotes do editor de texto Atom.

Ver também **atom**.

Mais informações: <https://github.com/atom/apm>.

- Instalar pacotes de <http://atom.io/packages> e temas de <http://atom.io/themes>:

```
apm install {{nome_do_pacote}}
```

- Remover pacotes/temas:

```
apm remove {{nome_do_pacote}}
```

- Atualizar pacotes/temas:

```
apm upgrade {{nome_do_pacote}}
```



# arch

Exibir o nome da arquitetura do sistema.

Veja também **uname**.

Mais informações: <https://www.gnu.org/software/coreutils/arch>.

- Exibir a arquitetura do sistema:

```
arch
```

# arduino-builder

Uma ferramenta de linha de comando para compilar sketches do arduino.

**AVISO DE OBSOLESCÊNCIA:** Esta ferramenta está sendo descontinuada e substituída pelo **arduino**.

Mais informações: <https://github.com/arduino/arduino-builder>.

- Compilar um sketch:

```
arduino-builder -compile {{caminho/para/sketch.ino}}
```

- Definir o nível de debug (1 a 10, o padrão é 5):

```
arduino-builder -debug-level {{nivel}}
```

- Definir um diretório de compilação customizado:

```
arduino-builder -build-path {{caminho/para/diretorio}}
```

- Usar um arquivo com as opções de compilação, em vez de especificar --hardware, --tools, etc. manualmente toda hora:

```
arduino-builder -build-options-file {{caminho/para/build.options.json}}
```

- Habilitar o modo verboso:

```
arduino-builder -verbose {{true}}
```

# arduino

Arduino Studio - Ambiente de Desenvolvimento Integrado para a plataforma Arduino.

Mais informações: <https://github.com/arduino/Arduino/blob/master/build/shared/manpage.adoc>.

- Compilar um sketch:

```
arduino --verify {{caminho/para/arquivo.ino}}
```

- Compilar e enviar sketch:

```
arduino --upload {{caminho/para/arquivo.ino}}
```

- Compilar e enviar sketch para um Arduino Nano com uma CPU Atmega328p, conectada na porta /dev/ttyACM0:

```
arduino --board {{arduino:avr:nano:cpu=atmega328p}} --port {{/dev/ttyACM0}} --upload {{caminho/para/arquivo.ino}}
```

- Definir a preferência nome para um determinado valor:

```
arduino --pref {{nome}}={{valor}}
```

- Compilar um sketch, colocar o resultado da compilação no diretório de compilação, e reutilizar qualquer resultado pre-existente neste diretório:

```
arduino --pref build.path={{caminho/para/diretório}} --verify {{caminho/para/arquivo.ino}}
```

- Salvar todas as preferências (alteradas) para preferences.txt:

```
arduino --save-prefs
```

# arp-scan

Envia pacotes ARP para máquinas (identificadas por endereço IP ou por nome de domínio) em uma rede local, identificando as máquinas ativas de acordo com as respostas.

Mais informações: <https://github.com/royhills/arp-scan>.

- Verificar as máquinas da rede local:

```
arp-scan --localnet
```

- Verificar as máquinas de uma rede IP especificando a máscara de bit:

```
arp-scan {{192.168.1.1}}/{{24}}
```

- Verificar as máquinas de uma rede IP que estejam em uma faixa de valores:

```
arp-scan {{127.0.0.0}}-{{127.0.0.31}}
```

- Verificar as máquinas de uma rede IP especificando a máscara de rede:

```
arp-scan {{10.0.0.0}}:{{255.255.255.0}}
```

# arp

Mostrar e manipular a cache ARP do sistema.

Mais informações: <https://manned.org/arp>.

- Mostrar a tabela arp atual:

```
arp -a
```

- Eliminar uma entrada específica:

```
arp -d {{endereço}}
```

- Criar uma entrada:

```
arp -s {{endereço}} {{endereço_mac}}
```

# asciidoctor

Um processador que converte AsciiDoc em formatos publicáveis.

Mais informações: <https://docs.asciidoctor.org>.

- Converte um arquivo .adoc em HTML (formato padrão de saída):

```
asciidoctor {{caminho/para/arquivo.adoc}}
```

- Converte um arquivo .adoc em HTML e liga a uma folha de estilos CSS:

```
asciidoctor -a stylesheet={{caminho/para/estilos.css}}  
{{caminho/para/arquivo.adoc}}
```

- Converte um arquivo .adoc em um HTML embutível, removendo tudo exceto o corpo:

```
asciidoctor --embedded {{caminho/para/arquivo.adoc}}
```

- Converte um arquivo .adoc em PDF usando a biblioteca asciidoctor-pdf:

```
asciidoctor --backend={{pdf}} --require={{asciidoctor-pdf}}  
{{caminho/para/arquivo.adoc}}
```

# asdf

Utilitário para a gestão de versões de linguagens e programas.

Mais informações: <https://asdf-vm.com>.

- Listar todos os plugins disponíveis:

```
asdf plugin-list-all
```

- Instalar um plugin:

```
asdf plugin-add {{nome}}
```

- Listar todas as versões disponíveis para um pacote:

```
asdf list-all {{nome}}
```

- Instalar uma versão específica de um pacote:

```
asdf install {{nome}} {{versão}}
```

- Definir a versão global de um pacote:

```
asdf global {{nome}} {{versão}}
```

- Definir a versão local de um pacote:

```
asdf local {{nome}} {{versão}}
```

# at

Ferramenta para o agendamento de comandos.

O serviço atd (ou atrun) deve estar sendo executado para as atuais execuções.

Mais informações: <https://manned.org/at>.

- Executar comandos da standard input em 5 minutos (pressionar `Ctrl + D` quando acabar):

```
at now + {{5}} minutes
```

- Executar um comando da standard input às 10:00 da manhã de hoje:

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

- Executar comandos de um dado arquivo na próxima terça:

```
at -f {{caminho/para/arquivo}} 9:30 PM Tue
```



# autossh

Executa, monitora e reinicia conexões SSH.

Reconecta automaticamente para manter os túneis de redirecionamento de porta ativos. Aceita todas as flags do **ssh**.

Mais informações: <https://www.harding.motd.ca/autossh>.

- Iniciar uma sessão SSH, reiniciando quando uma porta de monitoramento falhar em retornar dados:

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

- Redirecionar uma porta local para uma porta remota, reiniciando quando necessário:

```
autossh -M {{porta_de_monitoramento}} -L  
{{porta_local}}:localhost:{{porta_remota}} {{usuário}}  
@{{host}}
```

- Executar o autossh em segundo plano antes de executar o ssh e não abrir um shell remoto:

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

- Executar em segundo plano, sem porta de monitoramento, e em vez disso enviar pacotes de keep-alive SSH a cada 10 segundos para detectar falhas:

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

- Executar em segundo plano, sem porta de monitoramento e sem shell remoto, saindo se a redireção da porta falhar:

```
autossh -f -M 0 -N -o "ServerAliveInterval 10" -o  
"ServerAliveCountMax 3" -o ExitOnForwardFailure=yes -L  
{{porta_local}}:localhost:{{porta_remota}} {{usuário}}  
@{{host}}
```

- Executar em segundo plano, registrando a saída de depuração do autossh e a saída detalhada do ssh em arquivos:

```
AUTOSSH_DEBUG=1 AUTOSSH_LOGFILE={{caminho/para/  
arquivo_de_log_do_autossh.log}} autossh -f -M  
{{porta_de_monitoramento}} -v -E {{caminho/para/  
arquivo_de_log_do_ssh.log}} {{comando_ssh}}
```

# awk

Uma linguagem de programação versátil para trabalhar com arquivos.

Mais informações: <https://github.com/onetrueawk/awk>.

- Imprime a quinta coluna (também chamada de campo) em um arquivo separado por espaços:

```
awk '{print $5}' {{nome_do_arquivo}}
```

- Imprime a segunda coluna das linhas contendo "foo" em um arquivo separado por espaços:

```
awk '/{{foo}}/ {print $2}' {{nome_do_arquivo}}
```

- Imprime a última coluna de cada linha em um arquivo, usando vírgula (ao invés de espaço) como separador de campo:

```
awk -F ',' '{print $NF}' {{nome_do_arquivo}}
```

- Soma os valores da primeira coluna de um arquivo e imprime o total:

```
awk '{s+=$1} END {print s}' {{nome_do_arquivo}}
```

- Imprime de três em três linhas a partir da primeira:

```
awk 'NR%3==1' {{nome_do_arquivo}}
```

- Imprime diferentes valores baseado em condições:

```
awk '{if ($1 == "foo") print "Correspondência completa foo";  
else if ($1 ~ "bar") print "Correspondência parcial bar";  
else print "Baz"}' {{nome_do_arquivo}}
```

- Imprime todas as linhas em que a 10ª coluna é igual a um dado valor:

```
awk '($10 == valor)'
```

- Imprime todas as linhas em que o valor da décima coluna está entre um mínimo e um máximo:

```
awk '($10 >= valor_minimo && $10 <= valor_maximo)'
```

# aws codeartifact

Interface de linha de comando para o AWS CodeArtifact.

O CodeArtifact permite armazenar artefatos usando gerenciadores de pacotes populares e criar ferramentas como Maven, Gradle, npm, Yarn, Twine, pip e NuGet.

Mais informações: <https://awscli.amazonaws.com/v2/documentation/api/latest/reference/codeartifact/index.html>.

- Lista domínios disponíveis para a sua conta da AWS:

```
aws codeartifact list-domains
```

- Gera credenciais para um gerenciador de pacote específico (p.e.: npm, pip):

```
aws codeartifact login --tool {{gerenciador_de_pacotes}} --  
domain {{seu_domínio}} --repository {{nome_do_repositório}}
```

- Recupera a URL do endpoint de um repositório do CodeArtifact:

```
aws codeartifact get-repository-endpoint --domain  
{{seu_domínio}} --repository {{nome_do_repositório}} --format  
{{npm|pypi|maven|nuget|generic}}
```

- Lista todos os comandos disponíveis para o CodeArtifact:

```
aws codeartifact help
```

- Exibe ajuda específica para um subcomando do CodeArtifact:

```
aws ec2 {{subcommand}} help
```

# aws configure

Gerencia as configurações para o AWS CLI.

Mais informações: <https://docs.aws.amazon.com/cli/latest/reference/configure/>.

- Configura interativamente o AWS CLI (cria uma nova configuração ou atualiza a configuração default):

```
aws configure
```

- Configura interativamente um profile para o AWS CLI (cria um novo profile ou atualiza um que já existae):

```
aws configure --profile {{nome_do_profile}}
```

- Exibe o valor de uma variável específica de configuração:

```
aws configure get {{nome}}
```

- Exibe o valor de uma variável específica de configuração de um profile específico:

```
aws configure get {{nome}} --profile {{nome_do_profile}}
```

- Altera o valor de uma variável específica de configuração:

```
aws configure set {{nome}} {{valor}}
```

- Altera o valor de uma variável específica de configuração de um profile específico:

```
aws configure set {{nome}} {{valor}} --profile {{nome_do_profile}}
```

- Lista os entradas da configuração:

```
aws configure list
```

- Lista os entradas da configuração de um profile específico:

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

# aws cur

Cria, pesquisa e apaga relatórios de uso do AWS.

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

- Cria um relatório de uso e custo do AWS definido de a partir de um arquivo JSON:

```
aws cur put-report-definition --report-definition file://  
{{caminho/para/definição_do_relatório.json}}
```

- Lista as definições dos relatórios de uso para a conta logada:

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

- Apaga uma definição de relatório de uso:

```
aws cur --region {{região_aws}} delete-report-definition --  
report-name {{relatório}}
```

# aws ec2

Interface de linha de comando para o AWS EC2.

Provê capacidade computacional segura e flexível na nuvem da AWS para proporcionar um desenvolvimento e subida para produção de aplicações rapidamente.

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

- Exibe informações sobre uma instância específica:

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

- Exibe informações sobre todas as instâncias:

```
aws ec2 describe-instances
```

- Exibe informações sobre todos os volumes EC2:

```
aws ec2 describe-volumes
```

- Deleta um volume EC2:

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

- Cria um snapshot de um volume EC2:

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

- Lista as AMIs (Imagem de Máquina da Amazon) disponíveis:

```
aws ec2 describe-images
```

- Lista todos os comandos EC2 disponíveis:

```
aws ec2 help
```

- Exibe ajuda específica para um subcomando da EC2:

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

# aws ecr

Enviar, buscar, e gerenciar imagens de container.

Mais informações: <https://awscli.amazonaws.com/v2/documentation/api/latest/reference/ecr/index.html>.

- Autentica o docker com o registro default (nome do usuário na AWS):

```
aws ecr get-login-password --region {{region}} | {{docker login}} --username AWS --password-stdin {{aws_account_id}}.dkr.ecr.{{region}}.amazonaws.com
```

- Cria um repositório:

```
aws ecr create-repository --repository-name {{repository}} --image-scanning-configuration scanOnPush={{true|false}} --region {{region}}
```

- Cria uma tag em uma imagem local para o ECR:

```
docker tag {{nome_do_container}}:{{tag}} {{aws_account_id}}.dkr.ecr.{{region}}.amazonaws.com/{{nome_do_container}}:{{tag}}
```

- Envia uma imagem para um repositório:

```
docker push {{aws_account_id}}.dkr.ecr.{{region}}.amazonaws.com/{{nome_do_container}}:{{tag}}
```

- Busca a imagem de um repositório:

```
docker pull {{aws_account_id}}.dkr.ecr.{{region}}.amazonaws.com/{{nome_do_container}}:{{tag}}
```

- Apaga uma imagem de um repositório:

```
aws ecr batch-delete-image --repository-name {{repositório}} --image-ids imageTag={{latest}}
```

- Apaga um repositório:

```
aws ecr delete-repository --repository-name {{repositório}} --force
```

- Lista as imagens de um repositório:

```
aws ecr list-images --repository-name {{repositório}}
```

# aws glue

Linha de comando CLI para o AWS Glue.

Define o endpoint público para o serviço AWS Glue.

Mais informações: <https://docs.aws.amazon.com/cli/latest/reference/glue/>.

- Lista trabalhos:

```
aws glue list-jobs
```

- Inicia um trabalho:

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

- Inicia um fluxo de trabalho:

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

- Lista os gatilhos:

```
aws glue list-triggers
```

- Iniciar um gatilho:

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

- Cria um endpoint de desenvolvimento:

```
aws glue create-dev-endpoint --endpoint-name {{nome}} --role-  
arn {{papel_arn_usado_pelo_endpoint}}
```



# aws-google-auth

Linha de comando para obter credenciais (STS) temporárias AWS usando o Google Apps como um provedor (Single Sign-On) federado.

Mais informações: <https://github.com/cevoaustralia/aws-google-auth>.

- Loga com o Google SSO usando identificadores IDP e SP e criar credenciais com duração de uma hora:

```
aws-google-auth -u {{exemplo@example.com}} -I  
{{${G00GLE_IDP_ID}} -S {{${G00GLE_SP_ID}} -d {{3600}}
```

- Loga perguntando ([a]sking) qual papel usar (no caso de diversos papeis SAML disponíveis):

```
aws-google-auth -u {{exemplo@example.com}} -I  
{{${G00GLE_IDP_ID}} -S {{${G00GLE_SP_ID}} -d {{3600}} -a
```

- Resolve aliases para contas AWS:

```
aws-google-auth -u {{exemplo@example.com}} -I  
{{${G00GLE_IDP_ID}} -S {{${G00GLE_SP_ID}} -d {{3600}} -a --  
resolve-aliases
```

- Exibe informações de ajuda:

```
aws-google-auth -h
```

# aws help

Exibe ajuda sobre o AWS CLI.

Mais informações: <https://docs.aws.amazon.com/cli/latest/userguide/cli-usage-help.html>.

- Exibe a ajuda:

```
aws help
```

- Lista todos os tópicos disponíveis:

```
aws help topics
```

- Exibe ajuda sobre um tópico específico:

```
aws help {{nome_do_tópico}}
```

# aws history

Exibe o histórico dos comandos para o AWS CLI (o armazenamento do histórico dos comandos do AWS CLI deve estar habilitado).

Mais informações: <https://docs.aws.amazon.com/cli/latest/reference/history/>.

- Lista histórico dos comandos e seus IDs:

```
aws history list
```

- Exibe eventos relacionados a um comando específico dado um ID do comando:

```
aws history show {{id_do_comando}}
```

# aws iam

CLI for AWS IAM.

Mais informações: <https://awscli.amazonaws.com/v2/documentation/api/latest/reference/iam/index.html>.

- Exibe a página de ajuda do `aws iam` (incluindo todos os comandos iam disponíveis):

```
aws iam help
```

- Lista os usuários:

```
aws iam list-users
```

- Lista as políticas:

```
aws iam list-policies
```

- Lista os grupos:

```
aws iam list-groups
```

- Obtém os usuários de um grupo:

```
aws iam get-group --group-name {{nome_do_grupo}}
```

- Descreve uma política IAM:

```
aws iam get-policy --policy-arn arn:aws:iam::aws:policy/{{nome_da_política}}
```

- Lista as chaves de acesso:

```
aws iam list-access-keys
```

- Lista as chaves de acesso para um usuário específico:

```
aws iam list-access-keys --user-name {{nome_do_usuario}}
```

# aws kinesis

CLI oficial da AWS para o serviço de streamin de dados Amazon Kinesis.

Mais informações: <https://docs.aws.amazon.com/cli/latest/reference/kinesis/index.html#cli-aws-kinesis>.

- Lista todos os streams de uma conta:

```
aws kinesis list-streams
```

- Escreve um registro para um stream Kinesis:

```
aws kinesis put-record --stream-name {{nome}} --partition-key {{chave}} --data {{mensagem_codificaca_em_base64}}
```

- Escreve um registro para um stream Kinesis com codificação base64 inline:

```
aws kinesis put-record --stream-name {{nome}} --partition-key {{chave}} --data "$( echo "{{minha mensagem não codificada}}" | base64 )"
```

- Lista os shards disponíveis em um stream:

```
aws kinesis list-shards --stream-name {{nome}}
```

- Obtém uma iteração de shards para leitura da mensagem mais antiga no shard do stream:

```
aws kinesis get-shard-iterator --shard-iterator-type TRIM_HORIZON --stream-name {{nome}} --shard-id {{id}}
```

- Lê registros de um shard usando uma iteração de um shard:

```
aws kinesis get-records --shard-iterator {{iteração}}
```

# aws lambda

Linha de comando CLI para o AWS lambda.

Mais informações: <https://docs.aws.amazon.com/cli/latest/reference/lambda/>.

- Executa uma função:

```
aws lambda invoke --function-name {{nome}} {{caminho/para/a/resposta}}.json
```

- Executa uma função enviando um payload em formato JSON:

```
aws lambda invoke --function-name {{nome}} --payload {{json}} {{caminho/para/a/resposta}}.json
```

- Lista as funções:

```
aws lambda list-functions
```

- Exibe a configuração de uma função:

```
aws lambda get-function-configuration --function-name {{nome}}
```

- Lista os aliases de uma função:

```
aws lambda list-aliases --function-name {{nome}}
```

- Exibe o configuração de concorrência reservada de uma função:

```
aws lambda get-function-concurrency --function-name {{nome}}
```

- Lista quais serviços AWS pode invocar a função:

```
aws lambda get-policy --function-name {{nome}}
```

# aws s3 cp

Copia arquivos locais ou objetos do S3 para outros diretórios locais ou no S3.

Mais informações: <https://awscli.amazonaws.com/v2/documentation/api/latest/reference/s3/cp.html>.

- Copia um arquivo local para um bucket específico:

```
aws s3 cp {{caminho/do/arquivo}} s3://{{nome_do_bucket}}/
{{caminho/para/o/arquivo_remoto}}
```

- Copia um objeto específico para outro bucket dentro do S3:

```
aws s3 cp s3://{{nome_do_bucket1}}/{{caminho/do/arquivo}}
s3://{{nome_do_bucket2}}/{{caminho/para/o/destino}}
```

- Copia um objeto específico do S3 para outro bucket mantendo seu nome original:

```
aws s3 cp s3://{{nome_do_bucket1}}/{{caminho/do/arquivo}}
s3://{{nome_do_bucket2}}
```

- Copia objetos do S3 para um diretório local recursivamente:

```
aws s3 cp s3://{{nome_do_bucket}} . --recursive
```

- Exibe a ajuda:

```
aws s3 cp help
```

# aws s3

Interface de linha de comando para AWS S3.

Provê armazenamento através de uma interface de web services.

Mais informações: <https://awscli.amazonaws.com/v2/documentation/api/latest/reference/s3/index.html>.

- Exibe arquivos de um bucket:

```
aws s3 ls {{nome_do_bucket}}
```

- Sincroniza arquivos e diretórios locais para o bucket:

```
aws s3 sync {{caminho/para/arquivos}} s3://{{nome_do_bucket}}
```

- Sincroniza arquivos e diretórios do bucket para diretório local:

```
aws s3 sync s3://{{nome_do_bucket}} {{caminho/para/diretorio}}
```

- Sincroniza arquivos e diretórios excluindo algo:

```
aws s3 sync {{caminho/para/arquivos}} s3://{{nome_do_bucket}}  
--exclude {{arquivo/não/sincronizado}} --exclude {{caminho/  
não/sincronizado}}/*
```

- Remove arquivo do bucket:

```
aws s3 rm s3://{{nome_do_bucket}}/{{caminho/do/arquivo}}
```

- Somente exibe a prévia das mudanças:

```
aws s3 {{qualquer_comando}} --dryrun
```



# aws secretsmanager

Armazena, gerencia, e obtém secrets.

Mais informações: <https://docs.aws.amazon.com/cli/latest/reference/secretsmanager/>.

- Lista secrets armazenados pelo gerenciador de secretes na conta atual:

```
aws secretsmanager list-secrets
```

- Cria um secret:

```
aws secretsmanager create-secret --name {{nome}} --  
description "{{descrição_do_secret}}" --secret-string  
{{secret}}
```

- Apaga um secret:

```
aws secretsmanager delete-secret --secret-id {{nome_ou_arn}}
```

- Visualiza detalhes de um secret menos pelo texto do secret:

```
aws secretsmanager describe-secret --secret-id  
{{nome_ou_arn}}
```

- Obtém o valor do secret (para pegar a última versão do secret não use --version-stage):

```
aws secretsmanager get-secret-value --secret-id  
{{nome_ou_arn}} --version-stage {{versão_do_secret}}
```

- Alterna o secret imediatamente usando uma função Lambda:

```
aws secretsmanager rotate-secret --secret-id {{nome_ou_arn}}  
--rotation-lambda-arn {{arn_da_função_lambda}}
```

- Alterna o secret automaticamente a cada 30 dias usando uma função Lambda:

```
aws secretsmanager rotate-secret --secret-id {{nome_ou_arn}}  
--rotation-lambda-arn {{arn_da_função_lambda}} --rotation-  
rules AutomaticallyAfterDays={{30}}
```

# aws ses

Interface de linha de comando para o AWS Simple Email Service.

Serviço em nuvem com alta performance para envio e recebimento de emails.

Mais informações: <https://awscli.amazonaws.com/v2/documentation/api/latest/reference/ses/index.html>.

- Cria um novo conjunto de regras:

```
aws ses create-receipt-rule-set --rule-set-name  
{{nome_do_conjunto_de_regras}} --generate-cli-skeleton
```

- Descreve os conjuntos ativos de regras:

```
aws ses describe-active-receipt-rule-set --generate-cli-  
skeleton
```

- Descreve uma regra específica de um conjunto de regras:

```
aws ses describe-receipt-rule --rule-set-name  
{{nome_do_conjunto_de_regras}} --rule-name {{nome_da_regra}}  
--generate-cli-skeleton
```

- Lista todos os conjuntos de regras:

```
aws ses list-receipt-rule-sets --starting-token  
{{texto_do_token}} --max-items {{inteiro}} --generate-cli-  
skeleton
```

- Remove um conjunto de regras específico (o conjunto ativo não pode ser removido):

```
aws ses delete-receipt-rule-set --rule-set-name  
{{nome_do_conjunto_de_regras}} --generate-cli-skeleton
```

- Remove uma regra específica de um conjunto de regras:

```
aws ses delete-receipt-rule --rule-set-name  
{{nome_do_conjunto_de_regras}} --rule-name {{nome_da_regra}}  
--generate-cli-skeleton
```

- Envia um email:

```
aws ses send-email --from {{de_endereco}} --destination  
"ToAddresses={{para_endereco}}" --message  
"Subject={Data={{assunto}},Charset=utf8},Body={Text={Data={{corpo_email}}}}
```

- Exibe ajuda para um subcomando específico do SES:

```
aws ses {{subcomando}} help
```

# aws sqs

Cria, apaga, e envia mensagens para filas para o serviço AWS SQS.

Mais informações: <https://awscli.amazonaws.com/v2/documentation/api/latest/reference/sqs/index.html>.

- Lista todas as filas disponíveis:

```
aws sqs list-queues
```

- Exibe a URL de uma fila específica:

```
aws sqs get-queue-url --queue-name {{nome_da_fila}}
```

- Cria uma fila com atributos especificados em arquivo JSON:

```
aws sqs create-queue --queue-name {{nome_da_fila}} --  
attributes {{file://caminho/para/arquivos_de_atributos.json}}
```

- Envia mensagem específica para uma fila:

```
aws sqs send-message --queue-url https://sqs.  
{{regiao}}.amazonaws.com/{{nome_da_fila}} --message-body  
"{{corpo_da_mensagem}}" --delay-seconds {{inteiro}} --  
message-attributes {{file://caminho/para/  
arquivos_de_atributos.json}}
```

- Remove uma mensagem específica de uma fila:

```
aws sqs delete-message --queue-url {{https://url_da_fila}} --  
receipt-handle {{identificado_da_mensagem}}
```

- Remove uma fila específica:

```
aws sqs delete-queue --queue-url https://sqs.  
{{regiao}}.amazonaws.com/{{nome_da_fila}}
```

- Remove todas as mensagens de uma fila específica:

```
aws sqs purge-queue --queue-url https://sqs.  
{{regiao}}.amazonaws.com/{{nome_da_fila}}
```

- Habilita uma conta AWS específica para enviar mensagens para uma fila:

```
aws sqs add-permission --queue-url https://sqs.  
{{regiao}}.amazonaws.com/{{nome_da_fila}} --label  
{{nome_da_permissao}} --aws-account-ids {{id_da_conta}} --  
actions SendMessage
```

# aws sts

Serviço de Token de Segurança (STS) que permite solicitar credenciais temporárias para usuários (IAM) ou federados.

Mais informações: <https://docs.aws.amazon.com/cli/latest/reference/sts/>.

- Obtém credenciais temporárias para acessar recursos AWS específicos:

```
aws sts assume-role --role-arn {{arn_do_papel_aws}}
```

- Obtém um usuário IAM ou papel que foi usado para chamar a operação:

```
aws sts get-caller-identity
```

# aws

A interface de linha de comando oficial para o Amazon Web Services.

Passo-a-passo, Single Sign-On (SSO), autocompletar de recursos e opções de YAML somente na v2.

Alguns subcomandos como **aws s3** tem sua própria documentação de uso.

Mais informações: <https://aws.amazon.com/cli>.

- Configura a linha de comando da AWS:

```
aws configure wizard
```

- Configura a linha de comando da AWS usando o SSO:

```
aws configure sso
```

- Veja o texto de ajuda para o comando da AWS:

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

- Obtenha a informações da identidade usada (útil para analisar problemas de permissão):

```
aws sts get-caller-identity
```

- Lista recursos da AWS em uma região em yaml:

```
aws dynamodb list-tables --region {{sa-east-1}} --output yaml
```

- Usa prompt de comando para ajuda com o preenchimento:

```
aws iam create-user --cli-auto-prompt
```

- Usa um passo-a-passo interativo para um recurso da AWS:

```
aws dynamodb wizard {{nova-tabela}}
```

- Gera um arquivo esqueleto em JSON (útil para ser usado em infraestrutura como código):

```
aws dynamodb update-table --generate-cli-skeleton
```

# axel

Acelerador de downloads.

Suporta HTTP, HTTPS, e FTP.

Mais informações: <https://github.com/axel-download-accelerator/axel>.

- Fazer download de uma URL para um arquivo:

```
axel {{url}}
```

- Fazer download especificando o nome do arquivo de destino:

```
axel {{url}} -o {{nome_do_arquivo}}
```

- Fazer download usando múltiplas conexões:

```
axel -n {{número_de_conexões}} {{url}}
```

- Procurar por mirrors:

```
axel -S {{número_de_mirrors}} {{url}}
```

- Limitar velocidade de download (em bytes por segundo):

```
axel -s {{velocidade}} {{url}}
```

# az

A interface de linha de comando oficial do Microsoft Azure.

Mais informações: <https://learn.microsoft.com/cli/azure>.

- Fazer log in no Azure:

```
az login
```

- Gerenciar informações de assinatura do Microsoft Azure:

```
az account
```

- Listar todos os discos gerenciados pelo Azure:

```
az disk list
```

- Listar todas as máquinas virtuais do Azure:

```
az vm list
```

- Gerenciar os serviços do Azure Kubernetes:

```
az aks
```

- Gerenciar recursos de rede do Azure:

```
az network
```



# banner

Exibe o argumento inserido como um grande banner em arte ASCII.

Mais informações: <https://manned.org/banner>.

- Exibe uma mensagem como banner (aspas opcionais):

```
banner "{{0la Mundo}}"
```

- Exibe um banner com a largura de 50 caracteres:

```
banner -w {{50}} "{{0la Mundo}}"
```

- Exibe um banner com o texto inserido na entrada padrão:

```
banner
```

# base32

Codifica ou decodifica um arquivo ou a entrada padrão (**stdin**) de/para Base32, para a saída padrão (**stdout**).

Mais informações: <https://www.gnu.org/software/coreutils/base32>.

- Codifica um arquivo:

```
base32 {{nome_do_arquivo}}
```

- Decodifica um arquivo:

```
base32 --decode {{nome_do_arquivo}}
```

- Codifica a partir de **stdin**:

```
{{algum_comando}} | base32
```

- Decodifica a partir de **stdin**:

```
{{algum_comando}} | base32 --decode
```

# base64

Codifica ou decodifica um arquivo ou uma entrada padrão (**stdin**) de/para Base64, para uma saída padrão (**stdout**).

Mais informações: <https://www.gnu.org/software/coreutils/base64>.

- Codifica o conteúdo de um arquivo para base64 e grava o resultado em **stdout**:

```
base64 {{nome_arquivo}}
```

- Decodifica o conteúdo de um arquivo em base64 e grava o resultado em **stdout**:

```
base64 --decode {{nome_arquivo}}
```

- Codifica a partir de **stdin**:

```
{{algum_comando}} | base64
```

- Decodifica a partir de **stdin**:

```
{{algum_comando}} | base64 --decode
```

# basename

Remove o prefixo de um caminho.

Mais informações: <https://www.gnu.org/software/coreutils/basename>.

- Mostra apenas o nome do arquivo em um caminho:

```
basename {{caminho/para/arquivo}}
```

- Mostra apenas o nome do diretório mais à direita em um caminho:

```
basename {{caminho/para/diretório/}}
```

- Mostra apenas o nome do arquivo em um caminho, com um sufixo removido:

```
basename {{caminho/para/arquivo}} {{sufixo}}
```

# bash

Bourne-Again SHell, um interpretador de linha de comando compatível com **sh**.

Veja também: **zsh**, **histexpand** (expansão do histórico).

Mais informações: <https://gnu.org/software/bash/>.

- Inicia uma sessão interativa do shell:

```
bash
```

- Inicia uma sessão interativa do shell sem carregar as configurações de inicialização:

```
bash --norc
```

- Executa [c]omandos específicos:

```
bash -c "{{echo 'bash é executado'}}"
```

- Executa um script específico:

```
bash {{caminho/para/script.sh}}
```

- Executa um script específico exibindo cada comando antes de executá-lo:

```
bash -x {{caminho/para/script.sh}}
```

- Executa um script específico e para no primeiro [e]rro:

```
bash -e {{caminho/para/script.sh}}
```

- Executa comandos específicos da stdin:

```
{{echo "echo 'bash é executado'"}} | bash
```

- Inicia uma sessão do shell [r]estrita:

```
bash -r
```

# bat

Imprime e concatena arquivos.

Um clone do **cat** com realce de sintaxe e integração com Git.

Mais informações: <https://github.com/sharkdp/bat>.

- Imprime o conteúdo de um arquivo para a saída padrão:

```
bat {{arquivo}}
```

- Concatena o conteúdo de vários arquivos em um arquivo destino:

```
bat {{arquivo1}} {{arquivo2}} > {{arquivo_destino}}
```

- Acrescenta o conteúdo de vários arquivos ao final de um arquivo destino:

```
bat {{arquivo1}} {{arquivo2}} >> {{arquivo_destino}}
```

- Numera todas as linhas de saída:

```
bat -n {{arquivo}}
```

- Realça a sintaxe de um arquivo JSON:

```
bat --language json {{arquivo.json}}
```

- Mostra todas as linguagens suportadas:

```
bat --list-languages
```

# bg

Retomar a execução, em segundo plano, de processos que foram suspensos (e.g. utilizando **Ctrl + Z**).

Mais informações: <https://manned.org/bg>.

- Retomar a execução, em segundo plano, do processo que foi suspenso mais recentemente:

```
bg
```

- Retomar a execução, em segundo plano, de um processo específico (utilizando `jobs -l` para obter o seu ID):

```
bg {{id_processo}}
```

# binwalk

Ferramenta de análise de Firmware.

Mais informações: <https://github.com/ReFirmLabs/binwalk>.

- Escaneia um arquivo binário:

```
binwalk {{caminho/para/binário}}
```

- Extrai arquivos de um binário, especificando a saída do diretório:

```
binwalk --extract --directory {{diretório_do_destino}}  
{{caminho/para/binário}}
```

- Extrai recursivamente arquivos de um binário limitando a profundidade da recursão para 2:

```
binwalk --extract --matryoshka --depth {{2}} {{caminho/para/  
binário}}
```

- Extrai arquivos de um binário com uma assinatura específica:

```
binwalk --dd '{{png image:png}}' {{caminho/para/binário}}
```

- Analisa a entropia de um binário, salvando o gráfico com o mesmo nome que o binário e a extensão .png:

```
binwalk --entropy --save {{caminho/para/binário}}
```

- Combina entropia, assinatura e análise dos código de operações em um comando só:

```
binwalk --entropy --signature --opcodes {{caminho/para/  
binário}}
```



# brew

A versão Linux do gerenciador de pacotes Homebrew.

Mais informações: <https://docs.brew.sh/Manpage>.

- Buscar por fórmulas disponíveis:

```
brew search {{termo_da_busca}}
```

- Instalar a última versão estável de uma fórmula (utilizar `--dev` para versões de desenvolvimento):

```
brew install {{formula}}
```

- Listar as fórmulas instaladas:

```
brew list
```

- Atualizar uma fórmula instalada (se não for informado o nome de uma fórmula, todas as fórmulas serão atualizadas):

```
brew upgrade {{formula}}
```

- Recuperar a versão mais recente do Linuxbrew e de todas as fórmulas do GitHub:

```
brew update
```

- Exibir as fórmulas que possuem novas versões disponíveis:

```
brew outdated
```

- Exibir informações sobre uma fórmula (versão, caminho de instalação, dependências, etc.):

```
brew info {{formula}}
```

- Verificar a instalação local em busca de possíveis problemas:

```
brew doctor
```

# btm

Uma alternativa ao **top**.

Tem como objetivo ser leve, multiplataforma e mais gráfico que o **top**.

Mais informações: <https://github.com/ClementTsang/bottom>.

- Exibe o layout padrão (CPU, memória, temperaturas, disco, rede e processos):

```
btm
```

- Ativa o modo básico, removendo gráficos e condensando dados (semelhante a **top**):

```
btm --basic
```

- Usa pontos grandes em vez de pequenos em gráficos:

```
btm --dot_marker
```

- Exibe também a carga da bateria e o estado de saúde:

```
btm --battery
```

- Atualiza a cada 250 milissegundos e exibe os últimos 30 segundos nos gráficos:

```
btm --rate 250 --default_time_value 30000
```

# bundle

Gerenciador de dependências da linguagem de programação Ruby.

Mais informações: <https://bundler.io/man/bundle.1.html>.

- Instalar todas as gemas definidas no `Gemfile`:

```
bundle install
```

- Atualizar todas as gemas, respeitando as regras definidas no `Gemfile`, e recriar o arquivo `Gemfile.lock`:

```
bundle update
```

- Atualizar uma gema específica definida no `Gemfile`:

```
bundle update --source {{nome_da_gema}}
```

- Criar o esqueleto do projeto de uma nova gema:

```
bundle gem {{nome_da_gema}}
```

# bundler

Este comando é um pseudônimo de **bundle**.

Mais informações: <https://bundler.io/man/bundle.1.html>.

- Ver documentação sobre o comando original:

```
tldr bundle
```

# bup

Sistema de backup baseado no formato Git packfile, oferecendo salvamentos incrementais e deduplicação global.

Mais informações: <https://github.com/bup/bup>.

- Inicializa um repositório de backup no diretório local especificado:

```
bup -d {{caminho/para/repositório}} init
```

- Prepara um determinado diretório antes de fazer um backup:

```
bup -d {{caminho/para/repositório}} index {{caminho/para/diretório}}
```

- Faz o backup de um diretório para o repositório:

```
bup -d {{caminho/para/repositório}} save -n  
{{nome_do_backup}} {{caminho/para/diretório}}
```

- Exibe os snapshots de backup armazenados atualmente no repositório:

```
bup -d {{caminho/para/repositório}} ls
```

- Restaura um snapshot de backup específico para um diretório de destino:

```
bup -d {{caminho/para/repositório}} restore -C {{caminho/  
para/diretório_de_destino}} {{nome_do_backup}}
```

# bzip2

Um compressor de arquivos que utiliza o algoritmo Burrows–Wheeler.

Mais informações: <https://manned.org/bzip2>.

- Compactar um arquivo:

```
bzip2 {{arquivo}}
```

- Descompactar um arquivo:

```
bzip2 -d {{arquivo.bz2}}
```

- Descompactar um arquivo exibindo o conteúdo no terminal:

```
bzip2 -dc {{arquivo.bz2}}
```

# c99

Compila programas em C de acordo com a norma ISO C.

Mais informações: <https://manned.org/c99>.

- Compila arquivo(s) de origem e cria um executável:

```
c99 {{arquivo.c}}
```

- Compila arquivo(s) de origem e cria um executável com um nome customizável:

```
c99 -o {{nome_executavel}} {{arquivo.c}}
```

- Compila arquivo(s) de origem e cria arquivo(s) de objeto:

```
c99 -c {{arquivo.c}}
```

- Compila arquivo(s) de origem, linka com arquivo(s) de objeto e cria um executável:

```
c99 {{arquivo.c}} {{arquivo.o}}
```

# cabal

Interface de linha de comando para a infraestrutura de pacote Haskell (Cabal).

Gerencia projetos Haskell e pacotes Cabal do repositório de pacotes Hackage.

Mais informações: <https://cabal.readthedocs.io/en/latest/intro.html>.

- Busca e lista pacotes do Hackage:

```
cabal list {{string_buscada}}
```

- Mostra informações sobre o pacote:

```
cabal info {{nome_pacote}}
```

- Baixa e instala um pacote:

```
cabal install {{nome_pacote}}
```

- Cria um novo projeto Haskell no diretório atual:

```
cabal init
```

- Monta o projeto no diretório atual:

```
cabal build
```

- Roda testes do projeto no diretório atual:

```
cabal test
```



# caddy

Um servidor web open source, pronto para empresas, com HTTPS automático, escrito em Go.

Mais informações: <https://caddyserver.com>.

- Inicia Caddy em primeiro plano:

```
caddy run
```

- Inicia Caddy com um arquivo Caddy específico:

```
caddy run --config {{caminho/para/arquivoCaddy}}
```

- Inicia Caddy no plano de fundo:

```
caddy start
```

- Para um processo Caddy em plano de fundo:

```
caddy stop
```

- Executa um servidor de arquivo simples na porta especificada, com uma interface navegável:

```
caddy file-server --listen :{{8000}} --browse
```

- Executa um servidor proxy reverso:

```
caddy reverse-proxy --from :{{80}} --to localhost:{{8000}}
```

# calc

Uma calculadora interativa de precisão arbitrária no terminal.

Mais informações: <https://github.com/lcn2/calc>.

- Inicia a `calc` no modo interativo:

```
calc
```

- Realiza um cálculo no modo não interativo:

```
calc '{{85 * (36 / 4)}}'
```

- Realiza um cálculo sem qualquer formatação de saída (para usar com pipes):

```
calc -p '{{4/3 * pi() * 5^3}}'
```

- Realiza um cálculo e, em seguida, altera para o modo [i]nterativo:

```
calc -i '{{sqrt(2)}}'
```

- Inicia `calc` em um [m]odo de permissão específico (0 até 7, o padrão é 7):

```
calc -m {{modo}}
```

- Exibe uma introdução à `calc`:

```
calc help intro
```

- Exibe uma visão geral da `calc`:

```
calc help overview
```

- Abre o manual da `calc`:

```
calc help
```

# calendar

Mostra eventos de um arquivo calendar.

Mais informações: <https://www.commandlinux.com/man-page/man1/calendar.1.html>.

- Mostra eventos para hoje e amanhã (ou para o final de semana na sexta-feira) do calendário padrão:

```
calendar
```

- Mostra eventos marcados para os próximos 30 dias ([A]head):

```
calendar -A {{30}}
```

- Mostra eventos ocorridos nos últimos 7 dias ([B]ack):

```
calendar -B {{7}}
```

- Mostra eventos de um calendário personalizado, salvo no caminho especificado ([f]ile):

```
calendar -f {{caminho/para/arquivo}}
```

# cargo add

Adiciona dependências ao arquivo **Cargo.toml** de um projeto Rust.

Mais informações: <https://doc.rust-lang.org/cargo/commands/cargo-add.html>.

- Adicionar a versão mais recente de uma dependência ao projeto atual:

```
cargo add {{dependência}}
```

- Adicionar uma versão específica de uma dependência:

```
cargo add {{dependência}}@{{versão}}
```

- Adicionar uma dependência e habilitar uma ou mais funcionalidades específicas:

```
cargo add {{dependência}} --features {{funcionalidade_1}},  
{{funcionalidade_2}}
```

- Adicionar uma dependência opcional, que será exposta como uma funcionalidade da crate:

```
cargo add {{dependência}} --optional
```

- Adicionar uma crate local como dependência:

```
cargo add --path {{caminho/para/crate}}
```

- Adicionar uma dependência de desenvolvimento ou de compilação:

```
cargo add {{dependência}} --{{dev|build}}
```

- Adicionar uma dependência com todas as funcionalidades padrão desabilitadas:

```
cargo add {{dependência}} --no-default-features
```

# cargo build

Compila um projeto Rust em um pacote local incluindo todas as suas dependências.

Mais informações: <https://doc.rust-lang.org/cargo/commands/cargo-build.html>.

- Compilar o pacote ou pacotes definidos pelo arquivo `Cargo.toml` no diretório local:

```
cargo build
```

- Compilar os artefatos em modo de publicação (release), com otimizações:

```
cargo build --release
```

- Compilar um pacote garantindo que o `Cargo.lock` esteja atualizado:

```
cargo build --locked
```

- Compilar todos os pacotes no workspace:

```
cargo build --workspace
```

- Compilar um pacote específico:

```
cargo build --package {{pacote}}
```

- Compilar apenas o binário especificado:

```
cargo build --bin {{nome}}
```

- Compilar apenas um teste específico:

```
cargo build --test {{nome_do_teste}}
```

# cargo clippy

Conjunto de validadores para identificar erros comuns e melhorar código em Rust.

Mais informações: <https://github.com/rust-lang/rust-clippy>.

- Executar verificações no código no diretório atual:

```
cargo clippy
```

- Executar verificações garantindo que o `Cargo.lock` esteja atualizado:

```
cargo clippy --locked
```

- Executar verificações em todos os pacotes no workspace:

```
cargo clippy --workspace
```

- Executar verificações para um pacote específico:

```
cargo clippy --package {{pacote}}
```

- Executar validações tratando avisos como erros:

```
cargo clippy -- --deny warnings
```

- Executar verificações e ignorar avisos:

```
cargo clippy -- --allow warnings
```

- Aplicar automaticamente as sugestões do Clippy:

```
cargo clippy --fix
```

# cargo doc

Constrói e visualiza a documentação de um pacote Rust, opcionalmente em modo offline.

Mais informações: <https://doc.rust-lang.org/cargo/commands/cargo-doc.html>.

- Construir a documentação padrão do pacote e mostrá-la no navegador:

```
cargo doc --open
```

- Construir a documentação sem acessar a rede:

```
cargo doc --offline
```

- Visualizar a documentação de um pacote específico:

```
cargo doc --open --package {{pacote}}
```

- Visualizar a documentação de um pacote específico sem acessar a rede:

```
cargo doc --open --offline --package {{pacote}}
```

# cargo rustc

Compila um pacote Rust.

Mais informações: <https://doc.rust-lang.org/cargo/commands/cargo-rustc.html>.

- Compilar o pacote ou pacotes definidos pelo arquivo `Cargo.toml` no diretório de trabalho atual:

```
cargo rustc
```

- Compilar os artefatos em modo de publicação (release), com otimizações:

```
cargo rustc --release
```

- Compilar com otimizações específicas para a arquitetura do CPU atual:

```
cargo rustc --release -- -C target-cpu=native
```

- Compilar com otimização de velocidade:

```
cargo rustc -- -C opt-level {{1|2|3}}
```

- Compilar com otimização de tamanho (z também desativa a vetorização de ciclos):

```
cargo rustc -- -C opt-level {{s|z}}
```

- Verificar se o pacote usa código com padrões inseguros de acesso à memória:

```
cargo rustc --lib -- -D unsafe-code
```

- Compilar um pacote específico:

```
cargo rustc --package {{pacote}}
```

- Compilar apenas o binário especificado:

```
cargo --bin {{nome}}
```



# cargo test

Executa os testes unitários e de integração de um pacote Rust.

Mais informações: <https://doc.rust-lang.org/cargo/commands/cargo-test.html>.

- Executar apenas os testes que contenham uma string específica em seus nomes:

```
cargo test {{nomedoteste}}
```

- Definir o número de casos de teste para execução simultânea:

```
cargo test -- --test-threads={{quantidade}}
```

- Executar os testes garantindo que o `Cargo.lock` esteja atualizado:

```
cargo test --locked
```

- Testar os artefatos em modo de publicação (release), com otimizações:

```
cargo test --release
```

- Testar todos os pacotes no workspace:

```
cargo test --workspace
```

- Executar testes para um pacote específico:

```
cargo test --package {{pacote}}
```

- Executar testes sem ocultar a saída das execuções dos testes:

```
cargo test -- --nocapture
```

# cargo

Gerencia projetos Rust e as dependências dos módulos (crates).

Alguns subcomandos como **cargo build** tem a sua própria documentação.

Mais informações: <https://doc.rust-lang.org/cargo>.

- Procura por crates:

```
cargo search {{string_procurada}}
```

- Instala uma crate:

```
cargo install {{nome_da_crate}}
```

- Lista as crates instaladas:

```
cargo install --list
```

- Cria um projeto Rust sendo binário ou uma biblioteca no diretório atual:

```
cargo init --{{bin|lib}}
```

- Cria um projeto Rust sendo binário ou uma biblioteca em um diretório específico:

```
cargo new {{caminho/para/directório}} --{{bin|lib}}
```

- Constrói o projeto Rust no diretório atual:

```
cargo build
```

- Constrói o projeto Rust no diretório atual utilizando o nightly compilador:

```
cargo +nightly build
```

- Constrói o projeto Rust utilizando um número específico de threads (padrão é o número de cores do CPU):

```
cargo build --jobs {{número_de_threads}}
```

# cat

Exibe e concatena o conteúdo de arquivos.

Mais informações: <https://www.gnu.org/software/coreutils/cat>.

- Exibe o conteúdo de um arquivo na `stdout`:

```
cat {{caminho/para/arquivo}}
```

- Concatena o conteúdo de vários arquivos em um arquivo de saída:

```
cat {{caminho/para/arquivo1 caminho/para/arquivo2 ...}} >  
{{caminho/para/arquivo_de_saída}}
```

- Anexa o conteúdo de vários arquivos ao final de um arquivo de saída:

```
cat {{caminho/para/arquivo1 caminho/para/arquivo2 ...}} >>  
{{caminho/para/arquivo_de_saída}}
```

- Copia o conteúdo de um arquivo em um arquivo de saída sem armazenamento em buffer:

```
cat -u {{/dev/tty12}} > {{/dev/tty13}}
```

- Escreve a `stdin` em um arquivo:

```
cat - > {{caminho/para/arquivo}}
```

# cd

Altera o diretório de trabalho atual.

Mais informações: <https://manned.org/cd>.

- Vai para o diretório especificado:

```
cd {{caminho/para/diretório}}
```

- Vai para o diretório pai do diretório atual:

```
cd ..
```

- Vai para o diretório home do usuário atual:

```
cd
```

- Vai para o diretório home de um usuário especificado:

```
cd ~{{nome_de_usuario}}
```

- Vai para o diretório escolhido anteriormente:

```
cd -
```

- Vai para o diretório raiz:

```
cd /
```

# certutil

Gerencie chaves e certificados em bancos de dados e tokens NSS.

Mais informações: <https://manned.org/certutil>.

- Cria um novo banco de dados de certificados:

```
certutil -N -d .
```

- Lista todos os certificados em um banco de dados:

```
certutil -L -d .
```

- Lista todas as chaves privadas em um banco de dados:

```
certutil -K -d . -f {{caminho/para/arquivo_de_senha.txt}}
```

- Importa o certificado assinado para o banco de dados dos solicitantes:

```
certutil -A -n "{{certificado_do_servidor}}" -t ",," -i  
{{caminho/para/arquivo.crt}} -d .
```

- Adiciona nomes de assunto a um determinado certificado:

```
certutil -S -f {{caminho/para/arquivo_de_senha.txt}} -d . -t  
",," -c "{{certificado_do_servidor}}" -n  
"{{nome_do_servidor}}" -g {{2048}} -s  
"CN={{nome_comum}},O={{organização}}"
```

# chmod

Muda a permissão de acesso de um arquivo ou diretório.

Mais informações: <https://www.gnu.org/software/coreutils/chmod>.

- Dá ao [u]suário dono de um arquivo o direito de e[x]ecutá-lo:

```
chmod u+x {{arquivo}}
```

- Dá ao [u]suário direitos para le[r] e [w]escrever em um arquivo/diretório:

```
chmod u+rw {{arquivo_ou_diretorio}}
```

- Remove direitos e[x]ecutáveis de um [g]rupo:

```
chmod g-x {{arquivo}}
```

- Dá a [a]todos os usuários direitos para le[r] e e[x]ecutar:

```
chmod a+rx {{arquivo}}
```

- Dá para [o]utros (que não estejam no grupo do proprietário do arquivo) os mesmos direitos que o [g]rupo:

```
chmod o=g {{arquivo}}
```

- Remove todos os direitos de [o]utros:

```
chmod o= {{arquivo}}
```

- Muda recursivamente as permissões, dando para [g]rupo e [o]utros a habilidade para [w]escrever:

```
chmod -R g+w,o+w {{diretorio}}
```

- Recursivamente concede a [a]todos os usuários permissões de leitu[r]a para arquivos e e[X]ecute permissões para sub-diretórios dentro de um diretório:

```
chmod -R a+rX {{diretorio}}
```

# chown

Muda o usuário e grupo donos de arquivos e diretórios.

Mais informações: <https://www.gnu.org/software/coreutils/chown>.

- Muda o usuário que é dono de um arquivo ou diretório:

```
chown {{usuario}} {{caminho/para/arquivo_ou_diretorio}}
```

- Muda o usuário e grupo que são donos de um arquivo/diretório:

```
chown {{usuario}}:{{grupo}} {{caminho/para/arquivo_ou_diretorio}}
```

- Recursivamente muda o dono de um diretório e seu conteúdo:

```
chown -R {{usuario}} {{caminho/para/diretorio}}
```

- Muda o dono de um link simbólico:

```
chown -h {{usuario}} {{caminho/para/symlink}}
```

- Muda o dono de um arquivo/diretório para ficar igual a um arquivo de referência:

```
chown --reference={{caminho/para/arquivo_de_referencia}}  
{{caminho/para/arquivo_ou_diretorio}}
```

# chromium

Navegador código aberto do Google.

Mais informações: <https://www.chromium.org/developers/how-tos/run-chromium-with-flags/>.

- Abre um arquivo:

```
chromium {{caminho/para/arquivo.html}}
```

- Abre uma URL:

```
chromium {{exemplo.com}}
```

- Abre no modo de navegação anônima (incognito):

```
chromium --incognito {{exemplo.com}}
```

- Abre em uma nova janela:

```
chromium --new-window {{exemplo.com}}
```

- Abre no modo app (Sem barra de tarefas, barra de URL, botões, etc.):

```
chromium --app='{{https://example.com}}'
```

- Usa um servidor proxy:

```
chromium --proxy-server="{{socks5://hostname:66}}"  
{{exemplo.com}}
```



# clamav

Este comando é um pseudônimo de **clamscan**.

Mais informações: <https://www.clamav.net>.

- Ver documentação sobre o comando original:

```
tldr clamscan
```

# clang-cpp

Este comando é um pseudônimo de **clang++**.

- Ver documentação sobre o comando original:

```
tldr clang++
```

# clear

Limpa a tela do terminal.

Mais informações: <https://manned.org/clear>.

- Limpar a tela (equivalente a apertar Control-L no terminal Bash):

```
clear
```

- Limpar a tela mantendo o buffer de rolagem do terminal:

```
clear -x
```

- Especificar o tipo de terminal a ser limpo (por padrão é o valor da variável de ambiente `TERM`):

```
clear -T {{tipo_do_terminal}}
```

- Mostra a versão do `ncurses` usado pelo `clear`:

```
clear -V
```

# clj

Ferramenta de Clojure para iniciar um REPL ou invocar uma função com argumentos.

Todas as opções podem ser definidas em um arquivo **deps.edn**.

Mais informações: [https://clojure.org/guides/deps\\_and\\_cli](https://clojure.org/guides/deps_and_cli).

- Inicia um REPL:

```
clj
```

- Executa uma função:

```
clj -X {{namespace/function_name}}
```

- Executa a função principal (main) do namespace especificado:

```
clj -M -m {{namespace}} {{args}}
```

- Prepara um projeto resolvendo dependências, baixando bibliotecas, e criando / cacheando classpaths:

```
clj -P
```

- Inicia um servidor nREPL com o middleware CIDER:

```
clj -Sdeps '[:deps {nrepl {:mvn/version "0.7.0"} cider/cider-nrepl {:mvn/version "0.25.2"}}]' -m nrepl.cmdline --middleware '["cider.nrepl/cider-middleware"]' --interactive
```

- Inicia um REPL para ClojureScript e abre um navegador web:

```
clj -Sdeps '[:deps {org.clojure/clojurescript {:mvn/version "1.10.758"}}]' --main cljs.main --repl
```

# clojure

Este comando é um pseudônimo de **clj**.

- Ver documentação sobre o comando original:

```
tldr clj
```

# cmatrix

Exibe um padrão semelhante à Matrix rolando na tela do terminal.

Mais informações: <https://github.com/abishekvashok/cmatrix>.

- Habilita rolagem assíncrona:

```
cmatrix -a
```

- Exibe texto em vermelho:

```
cmatrix -C {{red}}
```

- Habilita modo arco-íris:

```
cmatrix -r
```

- Determina um atraso de 2 centissegundos (20 milissegundos) na atualização da tela:

```
cmatrix -u {{2}}
```

# code

Editor de código extensível e multi plataforma.

Mais informações: <https://github.com/microsoft/vscode>.

- Inicia Visual Studio Code:

```
code
```

- Abre arquivos/diretórios específicos:

```
code {{caminho/para/arquivo_ou_diretório1 caminho/para/arquivo_ou_diretório2 ...}}
```

- Compara dois arquivos específicos:

```
code --diff {{caminho/para/arquivo1}} {{caminho/para/arquivo2}}
```

- Abre arquivos/diretórios específicos em uma nova janela:

```
code --new-window {{caminho/para/arquivo_ou_diretório1 caminho/para/arquivo_ou_diretório2 ...}}
```

- Instala/desinstala uma extensão específica:

```
code --{{install|uninstall}}-extension {{editor.extensão}}
```

- Imprime as extensões instaladas:

```
code --list-extensions
```

- Imprime extensões instaladas com suas versões:

```
code --list-extensions --show-versions
```

- Inicia o editor como um superusuário (root) enquanto armazena dados do usuário em um diretório específico:

```
sudo code --user-data-dir {{caminho/para/diretório}}
```

# coffee

Executa scripts CoffeeScript ou os compila em JavaScript.

Mais informações: <https://coffeescript.org/cli>.

- Executa um script:

```
coffee {{caminho/para/arquivo.coffee}}
```

- Compila para JavaScript e salva em um arquivo com o mesmo nome:

```
coffee --compile {{caminho/para/arquivo.coffee}}
```

- Compila para JavaScript e salva em um arquivo de saída indicado:

```
coffee --compile {{caminho/para/arquivo.coffee}} --output  
{{caminho/para/arquivo.js}}
```

- Inicia um REPL (shell interativo):

```
coffee --interactive
```

- Observa script para alterações e o executa novamente:

```
coffee --watch {{caminho/para/arquivo.coffee}}
```



# cola

Este comando é um pseudônimo de **git-cola**.

- Ver documentação sobre o comando original:

```
tldr git-cola
```

# command

Obriga o shell a executar o programa, ignorando qualquer função ou alias com o mesmo nome.

Mais informações: <https://manned.org/command>.

- Executar o programa ls, mesmo que exista algum alias ls:

```
command {{ls}}
```

# compgen

Um programa para auto completar comandos no Bash, ele é executado ao pressionar duas vezes a tecla TAB.

Mais informações: <https://www.gnu.org/software/bash/manual/bash.html#index-compgen>.

- Exibir todos os comandos que você pode executar:

```
compgen -c
```

- Exibir todos os alias:

```
compgen -a
```

- Exibir todas as funções que você pode executar:

```
compgen -A function
```

- Exibir todas as palavras reservadas do shell:

```
compgen -k
```

- Exibir todos os comandos/alias que iniciam com o termo 'ls':

```
compgen -ac {{ls}}
```

# convert

Ferramenta de conversão de imagens da ImageMagick.

Mais informações: <https://imagemagick.org/script/convert.php>.

- Converter uma imagem do formato JPG para o formato PNG:

```
convert {{imagem.jpg}} {{imagem.png}}
```

- Escalar uma imagem para 50% do seu tamanho original:

```
convert {{imagem.png}} -resize 50% {{nova_imagem.png}}
```

- Escalar uma imagem, mantendo as suas proporções originais, para uma dimensão máxima de 640x480:

```
convert {{imagem.png}} -resize 640x480 {{nova_imagem.png}}
```

- Juntar várias imagens horizontalmente:

```
convert {{imagem1.png}} {{imagem2.png}} {{imagem3.png}}  
+append {{nova_imagem.png}}
```

- Criar um GIF a partir de uma série de imagens, com um intervalo de 100ms entre elas:

```
convert {{imagem1.png}} {{imagem2.png}} {{imagem3.png}} -  
delay {{100}} {{nova_imagem.gif}}
```

- Criar uma nova imagem de tamanho 800x600 com apenas um fundo sólido vermelho:

```
convert -size {{800x600}} "xc:{{#ff0000}}" {{imagem.png}}
```

# cp

Copia arquivos e diretórios.

Mais informações: <https://www.gnu.org/software/coreutils/cp>.

- Copia um arquivo para outra localização:

```
cp {{caminho/para/arquivo_origem.ext}} {{caminho/para/arquivo_destino.ext}}
```

- Copia um arquivo em outro diretório, mantendo o nome do arquivo:

```
cp {{caminho/para/arquivo_origem.ext}} {{caminho/para/diretório_pai_destino}}
```

- Copia recursivamente o conteúdo de um diretório para outra localização (se o destino existe, o diretório é copiado dentro dele):

```
cp -R {{caminho/para/diretório_origem}} {{caminho/para/diretório_destino}}
```

- Copia um diretório recursivamente, em modo verboso (mostra arquivos que estão sendo copiados):

```
cp -vR {{caminho/para/diretório_origem}} {{caminho/para/diretório_destino}}
```

- Copia múltiplos arquivos de uma só vez para um diretório:

```
cp -t {{caminho/para/diretório_destino}} {{caminho/para/arquivo1 caminho/para/arquivo2 ...}}
```

- Copia arquivos de texto para outra localização, em modo interativo (pergunta ao usuário antes de sobrescrever):

```
cp -i {{*.txt}} {{caminho/para/diretório_destino}}
```

- Segue links simbólicos antes de copiar:

```
cp -L {{link}} {{caminho/para/diretório_destino}}
```

- Usa o primeiro argumento como o diretório destino (útil para xargs ... |  
cp -t <DIR\_DEST>):

```
cp -t {{caminho/para/diretório_destino}} {{caminho/para/arquivo_ou_diretório1 caminho/para/arquivo_ou_diretório2 ...}}
```

# cron

Este comando é um pseudônimo de **crontab**.

- Ver documentação sobre o comando original:

```
tldr crontab
```

# curl

Transfere dados entre o computador local e um servidor remoto.

Suporta a maioria dos protocolos de comunicação, incluindo HTTP, FTP e POP3.

Mais informações: <https://curl.se/docs/manpage.html>.

- Descarregar os conteúdos de um URL para um arquivo:

```
curl {{http://example.com}} --output {{arquivo}}
```

- Descarregar um arquivo, gravando o resultado sob o nome do arquivo indicado pelo URL:

```
curl --remote-name {{http://example.com/arquivo}}
```

- Descarregar um arquivo, seguindo redirecionamentos e automaticamente continuando transferências idênticas que tenham sido interrompidas:

```
curl --remote-name --location --continue-at - {{http://example.com/arquivo}}
```

- Enviar dados codificados por formulário (pedido POST do tipo `application/x-www-form-urlencoded`):

```
curl --data {'nome=maria'}} {{http://example.com/formulario}}
```

- Enviar um pedido com um cabeçalho adicional, usando um método HTTP personalizado:

```
curl --header {'X-Meu-Cabecalho: 123'}} --request {{PUT}} {{http://example.com}}
```

- Enviar dados no formato JSON, especificando o cabeçalho de tipo de conteúdo (`content-type`) apropriado:

```
curl --data {'{"nome":"maria"}'} --header {'Content-Type: application/json'} {{http://example.com/usuarios/123}}
```

- Passar ao pedido o nome de usuário e senha para autenticação no servidor:

```
curl -u usuario:senha {{http://example.com}}
```

- Passar ao pedido o certificado do cliente e a chave para um recurso, omitindo a validação do certificado:

```
curl --cert {{client.pem}} --key {{key.pem}} --insecure  
{{https://example.com}}
```



# cut

Recorta campos do **stdin** ou de arquivos.

Mais informações: <https://www.gnu.org/software/coreutils/cut>.

- Imprime um intervalo específico de caracteres/campos de cada linha:

```
{{comando}} | cut --{{characters|fields}}={{1|1,10|1-10|1-|-10}}
```

- Imprime um intervalo de cada linha com um delimitador específico:

```
{{comando}} | cut --delimiter="{{{,}}}" --{{fields}}={{1}}
```

- Imprime um intervalo de cada linha de um arquivo específico:

```
cut --{{characters}}={{1}} {{caminho/para/arquivo}}
```

# daps

DAPS é um programa de código aberto para transformar DocBook XML em formatos de saída como HTML ou PDF.

Mais informações: <https://opensuse.github.io/daps/doc/index.html>.

- Verifica se um arquivo DocBook XML é válido:

```
daps -d {{caminho/para/arquivo.xml}} validate
```

- Converte um arquivo DocBook XML para PDF:

```
daps -d {{caminho/para/arquivo.xml}} pdf
```

- Converte um arquivo DocBook XML em um único arquivo HTML:

```
daps -d {{caminho/para/arquivo.xml}} html --single
```

- Exibe ajuda:

```
daps --help
```

- Exibe versão:

```
daps --version
```

# darkhttpd

Servidor web Darkhttpd.

Mais informações: <https://unix4lyfe.org/darkhttpd>.

- Inicia o servidor servindo a raiz do documento especificada:

```
darkhttpd {{caminho/para/raiz_do_documento}}
```

- Inicia o servidor na porta especificada (porta 8080 por padrão se estiver executando como usuário não raiz):

```
darkhttpd {{caminho/para/raiz_do_documento}} --port {{porta}}
```

- Escuta apenas no endereço IP especificado (por padrão, o servidor escuta em todas as interfaces):

```
darkhttpd {{caminho/para/raiz_do_documento}} --addr  
{{endereço_de_ip}}
```

# dart

A ferramenta para gerenciar projetos Dart.

Mais informações: <https://dart.dev/tools/dart-tool>.

- Inicializa um novo projeto Dart em um diretório com o mesmo nome:

```
dart create {{nome_do_projeto}}
```

- Executa um arquivo Dart:

```
dart run {{caminho/para/arquivo.dart}}
```

- Baixa as dependências do projeto atual:

```
dart pub get
```

- Executa testes de unidade para o projeto atual:

```
dart test
```

- Atualiza as dependências de um projeto desatualizado para oferecer suporte à segurança nula:

```
dart pub upgrade --null-safety
```

- Compila um arquivo Dart para um binário nativo:

```
dart compile exe {{caminho/para/arquivo.dart}}
```

# dash

Debian Almquist Shell, uma implementação moderna e compatível com POSIX de **sh** (não compatível com Bash).

Mais informações: <https://manned.org/dash>.

- Inicia uma sessão de shell interativa:

```
dash
```

- Executa um comando e sai:

```
dash -c "{{comando}}"
```

- Executa um script:

```
dash {{caminho/para/script.sh}}
```

- Executa comandos de um script, imprimindo cada comando antes de executá-lo:

```
dash -x {{caminho/para/script.sh}}
```

- Executa comandos de um script, parando no primeiro erro:

```
dash -e {{caminho/para/script.sh}}
```

- Lê e executa comandos de `stdin`:

```
dash -s
```

# date

Define ou exibe a data do sistema.

Mais informações: <https://www.gnu.org/software/coreutils/date>.

- Exibe a data atual usando o formato padrão de localidade:

```
date +%c
```

- Exibe a data atual em UTC, usando o formato ISO 8601:

```
date -u +%Y-%m-%dT%H:%M:%SZ
```

- Exibe a data atual em Unix timestamp - segundos desde 00:00:00 UTC de 1 de janeiro de 1970 (Unix epoch):

```
date +%s
```

- Converte uma data especificada como Unix timestamp para o formato padrão:

```
date -d @{{1473305798}}
```

- Converte uma determinada data pra Unix timestamp:

```
date -d "{{2018-09-01 00:00}}" +%s --utc
```

- Exibe a data atual usando o formato RFC-3339 (YYYY-MM-DD hh:mm:ss TZ):

```
date --rfc-3339=s
```

- Define a data atual usando o formato MMDDhhmmYYYY.ss (YYYY e .ss são opcionais):

```
date {{093023592021.59}}
```

- Exibe o número da semana ISO atual:

```
date +%V
```

# dd

Converte e copia um arquivo.

Mais informações: <https://www.gnu.org/software/coreutils/dd>.

- Cria um USB drive bootável a partir de um arquivo isohybrid (como uma archlinux-xxx.iso) e mostra o progresso:

```
dd if={{arquivo.iso}} of=/dev/{{usb_drive}} status=progress
```

- Clona um drive para outro drive com 4 MiB block, ignora erros e mostra o progresso:

```
dd if=/dev/{{drive_fonte}} of=/dev/{{drive_destino}} bs=4M  
conv=noerror status=progress
```

- Gera um arquivo com 100 bytes aleatórios utilizando o kernel random driver:

```
dd if=/dev/urandom of={{arquivo_random}} bs=100 count=1
```

- Faz o benchmark da performance de escrita de um disco:

```
dd if=/dev/zero of={{arquivo_1GB}} bs=1024 count=1000000
```

- Gera um backup do sistema em um arquivo IMG e mostra o progresso:

```
dd if=/dev/{{dispositivo_drive}} of={{caminho/para/  
arquivo.img}} status=progress
```

- Restaura um drive a partir de um arquivo IMG e mostra o progresso:

```
dd if={{caminho/para/arquivo.img}} of=/dev/  
{{dispositivo_drive}} status=progress
```

- Checa o progresso de um processo dd rodando (rode esse comando de outro shell):

```
kill -USR1 $(pgrep ^dd)
```

# diff

Compara diretórios e arquivos.

Mais informações: <https://man7.org/linux/man-pages/man1/diff.1.html>.

- Compara arquivos (mostra as mudanças necessárias para transformar `arquivo_antigo` em `arquivo_novo`):

```
diff {{arquivo_antigo}} {{arquivo_novo}}
```

- Compara arquivos, ignorando espaço:

```
diff --ignore-all-space {{arquivo_antigo}} {{arquivo_novo}}
```

- Compara arquivos, mostrando diferenças lado a lado:

```
diff --side-by-side {{arquivo_antigo}} {{arquivo_novo}}
```

- Compara arquivos, mostrando as diferenças de forma padronizada como feito por `git diff`:

```
diff --unified {{arquivo_antigo}} {{arquivo_novo}}
```

- Compara diretórios recursivamente, mostrando nomes de diretórios e arquivos diferentes e listando as diferenças entre os arquivos:

```
diff --recursive {{arquivo_antigo}} {{arquivo_novo}}
```

- Compara diretórios, mostrando apenas os nomes dos arquivos diferentes:

```
diff --recursive --brief {{arquivo_antigo}} {{arquivo_novo}}
```

- Cria um arquivo patch para o Git a partir das diferenças entre dois arquivos, tratando arquivos ausentes como vazios:

```
diff --text --unified --new-file {{arquivo_antigo}}  
{{arquivo_novo}} > {{diferenca.patch}}
```



# dig

Utilitário de pesquisa de DNS.

Mais informações: <https://manned.org/dig>.

- Pesquisa o(s) IP(s) associados a um hostname (Registros A):

```
dig +short {{exemplo.com}}
```

- Obtém uma resposta detalhada para um determinado domínio (Registros A):

```
dig +noall +answer {{exemplo.com}}
```

- Consulta um tipo de registro DNS específico associado a um nome de domínio fornecido:

```
dig +short {{exemplo.com}} {{A|MX|TXT|CNAME|NS}}
```

- Obtém todos os tipos de registros para um nome de domínio fornecido:

```
dig {{exemplo.com}} ANY
```

- Especifica um servidor DNS alternativo para consultar:

```
dig @{{8.8.8.8}} {{exemplo.com}}
```

- Realiza uma busca reversa de DNS em um endereço de IP (Registro PTR):

```
dig -x {{8.8.8.8}}
```

- Encontra servidores de nomes autorizados para a região e exibe os registros SOA:

```
dig +nssearch {{exemplo.com}}
```

- Realiza consultas iterativas e exibe o caminho de rastreamento completo para resolver um nome de domínio:

```
dig +trace {{exemplo.com}}
```

# docker build

Cria uma imagem a partir de um Dockerfile.

Mais informações: <https://docs.docker.com/engine/reference/commandline/build/>.

- Cria uma imagem docker usando o Dockerfile no diretório atual:

```
docker build .
```

- Cria uma imagem docker a partir de um Dockerfile em uma URL específica:

```
docker build {{github.com/creack/docker-firefox}}
```

- Cria uma imagem docker e cria uma etiqueta para ela:

```
docker build --tag {{nome:etiqueta}} .
```

- Cria uma imagem docker sem contexto de criação:

```
docker build --tag {{nome:etiqueta}} - < {{Dockerfile}}
```

- Não usa o cache na criação da imagem:

```
docker build --no-cache --tag {{nome:etiqueta}} .
```

- Cria uma imagem docker usando um Dockerfile específico:

```
docker build --file {{Dockerfile}} .
```

- Cria uma imagem docker utilizando variáveis customizadas para a criação de imagens:

```
docker build --build-arg {{PROXY_D0_HTTP=http://  
10.20.30.2:1234}} --build-arg {{PROXY_D0_FTP=http://  
40.50.60.5:4567}} .
```

# docker commit

Criar uma nova imagem a partir das alterações em um contêiner.

Mais informações: <https://docs.docker.com/engine/reference/commandline/commit/>.

- Criar uma imagem a partir de um contêiner específico:

```
docker commit {{contêiner}} {{imagem}}:{{tag}}
```

- Aplicar uma instrução CMD do Dockerfile à imagem criada:

```
docker commit --change="CMD {{comando}}" {{contêiner}}  
{{imagem}}:{{tag}}
```

- Aplicar uma instrução ENV do Dockerfile à imagem criada:

```
docker commit --change="ENV {{nome}}={{valor}}" {{contêiner}}  
{{imagem}}:{{tag}}
```

- Criar uma imagem com um autor específico nos metadados:

```
docker commit --author="{{autor}}" {{contêiner}} {{imagem}}:  
{{tag}}
```

- Criar uma imagem com um comentário específico nos metadados:

```
docker commit --message="{{comentário}}" {{contêiner}}  
{{imagem}}:{{tag}}
```

- Criar uma imagem sem pausar o contêiner durante o commit:

```
docker commit --pause={{false}} {{contêiner}} {{imagem}}:  
{{tag}}
```

- Exibir ajuda:

```
docker commit --help
```

# docker compose

Executa e gerencia multi-containers de aplicações Docker.

Mais informações: <https://docs.docker.com/compose/reference/>.

- Lista todos os containers em execução:

```
docker compose ps
```

- Cria e inicia todos os containers em segundo plano usando um arquivo `docker-compose.yml` do seu diretório atual:

```
docker compose up --detach
```

- Inicia todos os containers. Se necessário, realiza um rebuild:

```
docker compose up --build
```

- Inicia todos os containers que estão usando um arquivo compose alternativo:

```
docker compose --file {{caminho/para/arquivo}} up
```

- Encerra todos os containers em execução:

```
docker compose stop
```

- Encerra e remove todos os containers, networks, imagens e volumes:

```
docker compose down --rmi all --volumes
```

- Segue os logs de todos os containers:

```
docker compose logs --follow
```

- Segue os logs de um container específico:

```
docker compose logs --follow {{nome_container}}
```

# docker container

Gerenciar contêineres Docker.

Mais informações: <https://docs.docker.com/engine/reference/commandline/container/>.

- Listar os contêineres Docker em execução:

```
docker container ls
```

- Iniciar um ou mais contêineres parados:

```
docker container start {{nome_do_contêiner1}}  
{{nome_do_contêiner2}}
```

- Encerrar um ou mais contêineres em execução:

```
docker container kill {{nome_do_contêiner}}
```

- Parar um ou mais contêineres em execução:

```
docker container stop {{nome_do_contêiner}}
```

- Pausar todos os processos em um ou mais contêineres:

```
docker container pause {{nome_do_contêiner}}
```

- Exibir informações detalhadas sobre um ou mais contêineres:

```
docker container inspect {{nome_do_contêiner}}
```

- Exportar o sistema de arquivos de um contêiner como um arquivo tar:

```
docker container export {{nome_do_contêiner}}
```

- Criar uma nova imagem a partir das alterações em um contêiner:

```
docker container commit {{nome_do_contêiner}}
```

# docker cp

Copia arquivos ou diretórios entre filesystems do host e container.

Mais informações: <https://docs.docker.com/engine/reference/commandline/cp>.

- Copia um arquivo ou diretório de um host para um container:

```
docker cp {{caminho/do/arquivo_ou_diretório_do_host}}  
{{nome_do_container}}:{{caminho/do/  
arquivo_ou_diretório_do_container}}
```

- Copia um arquivo ou diretório de um container para o host:

```
docker cp {{nome_do_container}}:{{caminho/do/  
arquivo_ou_diretório_do_container}} {{caminho/do/  
arquivo_ou_diretório_do_host}}
```

- Copia um diretório de um host para um container, seguindo o link simbólico (copia os arquivos linkados diretamente e não o link simbólico):

```
docker cp --follow-link {{caminho/do/  
arquivo_ou_diretório_linkado_do_host}} {{nome_do_container}}:  
{{caminho/do/arquivo_ou_diretório_do_container}}
```

# docker exec

Executar um comando em um contêiner Docker em execução.

Mais informações: <https://docs.docker.com/engine/reference/commandline/exec/>.

- Entrar em uma sessão de shell interativa em um contêiner em execução:

```
docker exec --interactive --tty {{nome_do_contêiner}} {{/bin/bash}}
```

- Executar um comando em segundo plano (detached) em um contêiner em execução:

```
docker exec --detach {{nome_do_contêiner}} {{comando}}
```

- Selecionar o diretório de trabalho para a execução de um determinado comando:

```
docker exec --interactive --tty --workdir {{caminho/para/diretório}} {{nome_do_contêiner}} {{comando}}
```

- Executar um comando em segundo plano em um contêiner existente, mas manter o `stdin` aberto:

```
docker exec --interactive --detach {{nome_do_contêiner}} {{comando}}
```

- Definir uma variável de ambiente em uma sessão Bash em execução:

```
docker exec --interactive --tty --env {{nome_da_variável}}={{valor}} {{nome_do_contêiner}} {{/bin/bash}}
```

- Executar um comando como um usuário específico:

```
docker exec --user {{usuário}} {{nome_do_contêiner}} {{comando}}
```

# docker image

Gerencia imagens do Docker.

Veja também **docker build**, **docker import** e **docker pull**.

Mais informações: <https://docs.docker.com/engine/reference/commandline/image/>.

- Lista imagens Docker locais:

```
docker image ls
```

- Exclui imagens Docker locais não utilizadas:

```
docker image prune
```

- Exclui todas as imagens não utilizadas (não apenas aquelas sem uma etiqueta):

```
docker image prune --all
```

- Mostra o histórico de uma imagem Docker local:

```
docker image history {{imagem}}
```



# docker images

Gerencia imagens Docker.

Mais informações: <https://docs.docker.com/engine/reference/commandline/images/>.

- Lista todas as imagens Docker:

```
docker images
```

- Lista todas as imagens Docker incluindo imagens intermedirárias:

```
docker images --all
```

- Lista no modo silencioso (somente IDs numéricos):

```
docker images --quiet
```

- Lista todas as imagens Docker não usadas por nenhum container:

```
docker images --filter dangling=true
```

- Lista imagens que contenham um substring no seu nome:

```
docker images "{{*nome*}}"
```

# docker inspect

Retorna informações de baixo nível sobre objetos do Docker.

Mais informações: <https://docs.docker.com/engine/reference/commandline/inspect/>.

- Exibir ajuda:

```
docker inspect
```

- Exibir informações sobre um contêiner, imagem ou volume usando um nome ou ID:

```
docker inspect {{contêiner|imagem|ID}}
```

- Exibir o endereço IP de um contêiner:

```
docker inspect --format='{{range .NetworkSettings.Networks}}  
{{.IPAddress}}{{end}}' {{contêiner}}
```

- Exibir o caminho para o arquivo de log do contêiner:

```
docker inspect --format='{{.LogPath}}' {{contêiner}}
```

- Exibir o nome da imagem do contêiner:

```
docker inspect --format='{{.Config.Image}}' {{contêiner}}
```

- Exibir as informações de configuração como JSON:

```
docker inspect --format='{{json .Config}}' {{contêiner}}
```

- Exibir todas as portas vinculadas:

```
docker inspect --format='{{range $p, $conf  
:= .NetworkSettings.Ports}} {{ $p }} -> {{(index $conf  
0).HostPort}} {{end}}' {{contêiner}}
```

# docker load

Carregar imagens do Docker a partir de arquivos ou **stdin**.

Mais informações: <https://docs.docker.com/engine/reference/commandline/load/>.

- Carregar uma imagem do Docker a partir do **stdin**:

```
docker load < {{caminho/para/arquivo_imagem.tar}}
```

- Carregar uma imagem do Docker a partir de um arquivo específico:

```
docker load --input {{caminho/para/arquivo_imagem.tar}}
```

- Carregar uma imagem do Docker a partir de um arquivo específico no modo silencioso:

```
docker load --quiet --input {{caminho/para/arquivo_imagem.tar}}
```

# docker login

Fazer login em um registro do Docker.

Mais informações: <https://docs.docker.com/engine/reference/commandline/login/>.

- Fazer login interativamente em um registro:

```
docker login
```

- Fazer login em um registro com um nome de usuário específico (será solicitada a senha):

```
docker login --username {{nome_de_usuario}}
```

- Fazer login em um registro com nome de usuário e senha:

```
docker login --username {{nome_de_usuario}} --password {{senha}} {{servidor}}
```

- Fazer login em um registro com a senha vinda do stdin:

```
echo "{{senha}}" | docker login --username {{nome_de_usuario}} --password-stdin
```

# docker logs

Exibe os logs dos containers.

Mais informações: <https://docs.docker.com/engine/reference/commandline/logs>.

- Exibe logs de um container:

```
docker logs {{nome_do_container}}
```

- Exibe logs de um container e segue exibindo:

```
docker logs -f {{nome_do_container}}
```

- Exibe as últimas 5 linhas:

```
docker logs {{nome_do_container}} --tail {{5}}
```

- Exibe logs e adiciona a informação de hora ao log:

```
docker logs -t {{nome_do_container}}
```

- Exibe logs até um certo ponto no tempo de execução do container (por exemplo: 23m, 10s, 2013-01-02T13:23:37):

```
docker logs {{nome_do_container}} --until {{tempo}}
```

# docker-machine

Criar e gerenciar máquinas que executam o Docker.

Mais informações: <https://docs.docker.com/machine/reference/>.

- Listar as máquinas Docker em execução no momento:

```
docker-machine ls
```

- Criar uma nova máquina Docker com um nome específico:

```
docker-machine create {{nome}}
```

- Obter o status de uma máquina:

```
docker-machine status {{nome}}
```

- Iniciar uma máquina:

```
docker-machine start {{nome}}
```

- Parar uma máquina:

```
docker-machine stop {{nome}}
```

- Inspecionar informações sobre uma máquina:

```
docker-machine inspect {{nome}}
```

# docker network

Criar e gerenciar redes do Docker.

Mais informações: <https://docs.docker.com/engine/reference/commandline/network/>.

- Listar todas as redes disponíveis e configuradas no daemon do Docker:

```
docker network ls
```

- Criar uma rede definida pelo usuário:

```
docker network create --driver {{nome_do_driver}}  
{{nome_da_rede}}
```

- Exibir informações detalhadas de uma lista separada por espaços de redes:

```
docker network inspect {{nome_da_rede}}
```

- Conectar um contêiner a uma rede usando um nome ou ID:

```
docker network connect {{nome_da_rede}} {{nome_do_contêiner|  
ID}}
```

- Desconectar um contêiner de uma rede:

```
docker network disconnect {{nome_da_rede}}  
{{nome_do_contêiner|ID}}
```

- Remover todas as redes não utilizadas (que não são referenciadas por nenhum contêiner):

```
docker network prune
```

- Remover uma lista separada por espaços de redes não utilizadas:

```
docker network rm {{nome_da_rede}}
```

# docker ps

Lista os containers Docker.

Mais informações: <https://docs.docker.com/engine/reference/commandline/ps/>.

- Lista containers docker em execução:

```
docker ps
```

- Lista todos containers docker (em execução e parados):

```
docker ps --all
```

- Lista os últimos containers criados (inclui todos os estados):

```
docker ps --latest
```

- Filtra os containers que contém uma substring no seu nome:

```
docker ps --filter="name={{nome}}"
```

- Filtra todos os containers que possuem uma imagem antepassada:

```
docker ps --filter "ancestor={{imagem}}:{{tag}}"
```

- Filtra containers que tenha o código de saída:

```
docker ps --all --filter="exited={{código}}"
```

- Filtra containers por estado (criado, execução, removendo, pausado, finalizado e morto):

```
docker ps --filter="status={{estado}}"
```

- Filtra containers que contenham um volume específico ou montado em um caminho específico:

```
docker ps --filter="volume={{caminho/para/diretório}}" --format "table {{.ID}}\t{{.Image}}\t{{.Names}}\t{{.Mounts}}"
```



# docker pull

Baixar imagens do Docker de um registro.

Mais informações: <https://docs.docker.com/engine/reference/commandline/pull/>.

- Baixar uma imagem específica do Docker:

```
docker pull {{imagem}}:{{tag}}
```

- Baixar uma imagem específica do Docker no modo silencioso:

```
docker pull --quiet {{imagem}}:{{tag}}
```

- Baixar todas as tags de uma imagem específica do Docker:

```
docker pull --all-tags {{imagem}}
```

- Baixar imagens do Docker para uma plataforma específica, por exemplo, linux/amd64:

```
docker pull --platform {{linux/amd64}} {{imagem}}:{{tag}}
```

- Exibir ajuda:

```
docker pull --help
```

# docker rmi

Remove uma ou mais imagens Docker.

Mais informações: <https://docs.docker.com/engine/reference/commandline/rmi/>.

- Exibe a ajuda:

```
docker rmi
```

- Remove uma ou mais imagens pelo seus nomes:

```
docker rmi {{imagem1 imagem2 ...}}
```

- Remove forçadamente uma imagem:

```
docker rmi --force {{imagem}}
```

- Remove uma imagem sem apagar suas imagens pais que não possuem tags:

```
docker rmi --no-prune {{imagem}}
```

# docker run

Executa um comando em um novo container Docker.

Mais informações: <https://docs.docker.com/engine/reference/commandline/run/>.

- Executa um comando em um novo container de uma imagem taguada:

```
docker run {{imagem:tag}} {{comando}}
```

- Executa um comando em um novo container em background e exibe o ID:

```
docker run --detach {{image}} {{command}}
```

- Executa um comando em um novo container que será removido após a execução em um modo interativo e com um terminal TTY:

```
docker run --rm --interactive --tty {{image}} {{command}}
```

- Executa um comando em um novo container com variáveis de ambiente:

```
docker run --env '{{variável}}={{valor}}' --env {{variável}}  
{{imagem}} {{comando}}
```

- Executa um comando em um novo container montando volumes nos caminhos específicos:

```
docker run --volume {{caminho/no/host_local}}:{{caminho/no/  
container}} {{imagem}} {{comando}}
```

- Executa um comando em um novo container e abre as portas para acesso:

```
docker run --publish {{porta_do_host_local}}:  
{{porta_do_container}} {{imagem}} {{comando}}
```

# docker save

Exportar uma ou mais imagens do Docker para um arquivo de arquivamento.

Mais informações: <https://docs.docker.com/engine/reference/commandline/save/>.

- Salvar uma imagem redirecionando stdout para um arquivo tar:

```
docker save {{imagem}}:{{tag}} > {{caminho/para/arquivo.tar}}
```

- Salvar uma imagem em um arquivo tar:

```
docker save --output {{caminho/para/arquivo.tar}} {{imagem}}:{{tag}}
```

- Salvar todas as tags da imagem:

```
docker save --output {{caminho/para/arquivo.tar}} {{nome_da_imagem}}
```

- Selecionar tags específicas de uma imagem para salvar:

```
docker save --output {{caminho/para/arquivo.tar}} {{nome_da_imagem:tag1 nome_da_imagem:tag2 ...}}
```

# docker secret

Gerenciar segredos do Docker swarm.

Mais informações: <https://docs.docker.com/engine/reference/commandline/secret/>.

- Criar um novo segredo a partir de stdin:

```
{{comando}} | docker secret create {{nome_do_segredo}} -
```

- Criar um novo segredo a partir de um arquivo:

```
docker secret create {{nome_do_segredo}} {{caminho/para/arquivo}}
```

- Listar todos os segredos:

```
docker secret ls
```

- Exibir informações detalhadas sobre um ou vários segredos em um formato amigável ao usuário:

```
docker secret inspect --pretty {{nome_do_segredo1  
nome_do_segredo2 ...}}
```

- Remover um ou mais segredos:

```
docker secret rm {{nome_do_segredo1 nome_do_segredo2 ...}}
```

# docker service

Gerenciar os serviços em um daemon do Docker.

Mais informações: <https://docs.docker.com/engine/reference/commandline/service/>.

- Listar os serviços em um daemon do Docker:

```
docker service ls
```

- Criar um novo serviço:

```
docker service create --name {{nome_do_serviço}} {{imagem}}:{{tag}}
```

- Exibir informações detalhadas de uma lista separada por espaços de serviços:

```
docker service inspect {{nome_do_serviço|ID}}
```

- Listar as tarefas de uma lista separada por espaços de serviços:

```
docker service ps {{nome_do_serviço|ID}}
```

- Escalar para um número específico de réplicas para uma lista separada por espaços de serviços:

```
docker service scale {{nome_do_serviço}}={{quantidade_de_réplicas}}
```

- Remover uma lista separada por espaços de serviços:

```
docker service rm {{nome_do_serviço|ID}}
```

# docker-slim

Analisar e otimizar imagens Docker.

Mais informações: <https://github.com/docker-slim/docker-slim>.

- Iniciar o DockerSlim no modo interativo:

```
docker-slim
```

- Analisar as camadas do Docker a partir de uma imagem específica:

```
docker-slim xray --target {{imagem:tag}}
```

- Verificar um Dockerfile:

```
docker-slim lint --target {{caminho/para/Dockerfile}}
```

- Analisar e gerar uma imagem Docker otimizada:

```
docker-slim build {{imagem:tag}}
```

- Exibir ajuda para um subcomando:

```
docker-slim {{subcomando}} --help
```

# docker start

Inicia um ou mais containers parados.

Mais informações: <https://docs.docker.com/engine/reference/commandline/start/>.

- Exibe a ajuda:

```
docker start
```

- Inicia um container docker:

```
docker start {{container}}
```

- Inicia um container, attachando ao terminal os sinais `stdout` e `stderr` e outros sinais:

```
docker start --attach {{container}}
```

- Inicia um ou mais containers com ID separados por espaço:

```
docker start {{container1 container2 ...}}
```



# docker stats

Exibe estatísticas dinâmicas de uso de recursos dos containers.

Mais informações: <https://docs.docker.com/engine/reference/commandline/stats/>.

- Exibe estatísticas atualizadas de todos os containers em execução:

```
docker stats
```

- Exibe estatísticas atualizadas de uma lista separada por espaço dos containers:

```
docker stats {{nome_do_container}}
```

- Altera o formato das colunas para exibir o uso da CPU em porcentagem:

```
docker stats --format "{{.Name}}:\t{{.CPUPerc}}"
```

- Exibe estatísticas para todos os containers (tanto em execução como parados):

```
docker stats --all
```

- Desabilita estatísticas atualizadas e só exibe o status naquele momento:

```
docker stats --no-stream
```

# docker swarm

Uma ferramenta de orquestração de contêineres.

Mais informações: <https://docs.docker.com/engine/swarm/>.

- Inicializar um cluster do Swarm:

```
docker swarm init
```

- Exibir o token para ingressar como gerenciador ou trabalhador:

```
docker swarm join-token {{worker|manager}}
```

- Ingressar um novo nó ao cluster:

```
docker swarm join --token {{token}} {{url_do_nó_gerenciador:2377}}
```

- Remover um trabalhador do Swarm (executado dentro do nó trabalhador):

```
docker swarm leave
```

- Exibir o certificado CA atual no formato PEM:

```
docker swarm ca
```

- Rotacionar o certificado CA atual e exibir o novo certificado:

```
docker swarm ca --rotate
```

- Alterar o período de validade dos certificados dos nós:

```
docker swarm update --cert-expiry {{horas}}h{{minutos}}m{{segundos}}s
```

# docker system

Gerenciar dados do Docker e exibir informações do sistema em todo o sistema.

Mais informações: <https://docs.docker.com/engine/reference/commandline/system/>.

- Mostrar ajuda:

```
docker system
```

- Mostrar o uso de disco do Docker:

```
docker system df
```

- Mostrar informações detalhadas sobre o uso de disco:

```
docker system df --verbose
```

- Remover dados não utilizados:

```
docker system prune
```

- Remover dados não utilizados criados há mais de um período específico no passado:

```
docker system prune --filter="until={{horas}}h{{minutos}}m"
```

- Exibir eventos em tempo real do daemon do Docker:

```
docker system events
```

- Exibir eventos em tempo real de contêineres transmitidos como JSON Lines válidos:

```
docker system events --filter 'type=container' --format '{{json .}}'
```

- Exibir informações em todo o sistema:

```
docker system info
```

# docker tag

Atribuir tags a imagens Docker existentes.

Mais informações: <https://docs.docker.com/engine/reference/commandline/tag/>.

- Atribuir um nome e tag a um ID de imagem específico:

```
docker tag {{id}} {{nome}}:{{tag}}
```

- Atribuir uma tag a uma imagem específica:

```
docker tag {{imagem}}:{{tag_atual}} {{imagem}}:{{nova_tag}}
```

- Exibir ajuda:

```
docker tag
```

# docker update

Atualizar a configuração de contêineres Docker.

Este comando não é suportado para contêineres Windows.

Mais informações: <https://docs.docker.com/engine/reference/commandline/update/>.

- Atualizar a política de reinicialização a ser aplicada quando um contêiner específico for encerrado:

```
docker update --restart={{always|no|on-failure|unless-stopped}} {{nome_do_contêiner}}
```

- Atualizar a política para reiniciar até três vezes um contêiner específico quando ele for encerrado com status de saída diferente de zero:

```
docker update --restart=on-failure:3 {{nome_do_contêiner}}
```

- Atualizar o número de CPUs disponíveis para um contêiner específico:

```
docker update --cpus {{quantidade}} {{nome_do_contêiner}}
```

- Atualizar o limite de memória em [M]egabytes para um contêiner específico:

```
docker update --memory {{limite}}M {{nome_do_contêiner}}
```

- Atualizar o número máximo de IDs de processos permitidos dentro de um contêiner específico (use -1 para ilimitado):

```
docker update --pids-limit {{quantidade}}  
{{nome_do_contêiner}}
```

- Atualizar a quantidade de memória em [M]egabytes que um contêiner específico pode trocar para o disco (use -1 para ilimitado):

```
docker update --memory-swap {{limite}}M {{nome_do_contêiner}}
```

# docker volume

Gerenciar volumes do Docker.

Mais informações: <https://docs.docker.com/engine/reference/commandline/volume/>.

- Criar um volume:

```
docker volume create {{nome_do_volume}}
```

- Criar um volume com um rótulo específico:

```
docker volume create --label {{rótulo}} {{nome_do_volume}}
```

- Criar um volume tmpfs com tamanho de 100 MiB e uid 1000:

```
docker volume create --opt {{type}}={{tmpfs}} --opt  
{{device}}={{tmpfs}} --opt {{o}}={{size=100m,uid=1000}}  
{{nome_do_volume}}
```

- Listar todos os volumes:

```
docker volume ls
```

- Remover um volume:

```
docker volume rm {{nome_do_volume}}
```

- Exibir informações sobre um volume:

```
docker volume inspect {{nome_do_volume}}
```

- Remover todos os volumes locais não utilizados:

```
docker volume prune
```

- Exibir ajuda para um subcomando:

```
docker volume {{subcomando}} --help
```

# docker

Gerenciador de containers e imagens Docker.

Alguns subcomandos como **docker run** tem sua própria documentação de uso.

Mais informações: <https://docs.docker.com/engine/reference/commandline/cli/>.

- Listar os containers Docker que se encontram em execução:

```
docker ps
```

- Listar todos os containers Docker:

```
docker ps -a
```

- Inicializar um container com um nome personalizado a partir de uma imagem:

```
docker run --name {{nome_container}} {{imagem}}
```

- Começar ou parar um container existente:

```
docker {{start|stop}} {{nome_container}}
```

- Extrair uma imagem a partir de um Docker Registry:

```
docker pull {{imagem}}
```

- Abrir um terminal dentro de um container em execução:

```
docker exec -it {{nome_container}} {{sh}}
```

- Remover um container parado:

```
docker rm {{nome_container}}
```

- Obter e acompanhar o histórico de um container:

```
docker logs -f {{nome_container}}
```

# doxygen

Um sistema de documentação para várias linguagens de programação.

Mais informações: <http://www.doxygen.nl>.

- Gerar um arquivo de configuração padrão (Doxyfile):

```
doxygen -g
```

- Gerar um arquivo de configuração, especificando o nome do arquivo de configuração:

```
doxygen -g {{caminho/para/arquivo_configuração}}
```

- Gerar documentação utilizando um arquivo de configuração existente:

```
doxygen {{caminho/para/arquivo_configuração}}
```



# du

Uso de disco: estima e sumariza o uso de espaço em disco de arquivos e diretórios.

Mais informações: <https://www.gnu.org/software/coreutils/du>.

- Lista os tamanhos dos diretórios e qualquer subdiretório, em uma unidade de tamanho (B/KiB/MiB):

```
du -{{b|k|m}} {{caminho/para/diretorio}}
```

- Lista os tamanhos dos diretórios e subdiretórios, em tamanho legível por humanos (isto é seleciona automaticamente a unidade de tamanho apropriada para o tamanho):

```
du -h {{caminho/para/diretorio}}
```

- Exibe o tamanho de um único diretório, em tamanho legível por humanos:

```
du -sh {{caminho/para/diretorio}}
```

- Lista em tamanho legível por humanos todos os arquivos e diretórios dentro de um diretório:

```
du -ah {{caminho/para/diretorio}}
```

- Lista em tamanho legível por humanos, até o nível N de profundidade um diretório e subdiretórios:

```
du -h --max-depth=N {{caminho/para/diretorio}}
```

- Lista em tamanho legível por humanos todos os arquivos `.jpg` dos subdiretórios do diretório atual, e exibe o total cumulativo no final:

```
du -ch {{*/*.jpg}}
```

# dua

Dua (Analisador de Uso de Disco) é uma ferramenta para análise conveniente do uso de disco dado um diretório.

Mais informações: <https://github.com/Byron/dua-cli>.

- Analisa um diretório específico:

```
dua {{caminho/para/diretorio}}
```

- Exibe o tamanho aparente ao invés do uso do disco:

```
dua --apparent-size
```

- Contabiliza arquivos hard-linked cada vez que eles são encontrados:

```
dua --count-hard-links
```

- Agrega o espaço em disco consumido de um ou mais diretórios ou arquivos:

```
dua aggregate
```

- Inicia a interface de usuário:

```
dua interactive
```

- Seleciona o formato para contagem de bytes:

```
dua --format {{metric|binary|bytes|GB|GiB|MB|MiB}}
```

- Escolhe o número de threads a serem usadas:

```
dua --threads {{numero}}
```

# duc

Duc é uma coleção de ferramentas para indexar, inspecionar e visualizar uso do disco. O duc mantém uma base de dados dos tamanhos acumulados dos diretórios do sistema de arquivos, permitindo buscas nessa base, ou a criação de gráficos elegantes.

Mais informações: <https://duc.zevv.nl/>.

- Indexa o diretório /usr, escrevendo a base de dados para o local default em ~/.duc.db:

```
duc index {{/usr}}
```

- Lista todos os arquivos e diretórios dentro do /usr/local, mostrando tamanho relativo dos arquivos em um [g]raph (gráfico):

```
duc ls -Fg {{/usr/local}}
```

- Lista todos os arquivos e diretórios dentro do /usr/local em uma visão de árvore recursiva:

```
duc ls -Fg -R {{/usr/local}}
```

- Inicia uma interface gráfica para o usuário explorar o sistema de arquivos exibindo o gráfico sunburst:

```
duc gui {{/usr}}
```

- Executa a interface de console ncurses para explorar o sistema de arquivos:

```
duc ui {{/usr}}
```

- Exporta as informações da base de dados:

```
duc info
```

# dust

Dust oferece uma visão geral de quais diretórios estão usando espaço em disco.

Mais informações: <https://github.com/bootandy/dust>.

- Exibe informações para o diretório atual:

```
dust
```

- Exibe informações para uma lista de diretórios separados por espaço:

```
dust {{caminho/para/diretório1}} {{caminho/para/diretório2}}
```

- Exibe 30 diretórios (o padrão é 21):

```
dust --number-of-lines {{30}}
```

- Exibe informações para o diretório atual, com até 3 níveis de profundidade:

```
dust --depth {{3}}
```

- Exibe os maiores diretórios no topo em ordem decrescente:

```
dust --reverse
```

- Ignora todos os arquivos e diretórios com um nome específico:

```
dust --ignore-directory {{arquivo_ou_nome_do_diretório}}
```

- Não exibe barras de porcentagem e porcentagens:

```
dust --no-percent-bars
```

# dvc add

Adiciona um arquivo modificado para o índice.

Mais informações: <https://dvc.org/doc/command-reference/add>.

- Adiciona um arquivo para o índice:

```
dvc add {{caminho/para/arquivo}}
```

- Adiciona um diretório para o índice:

```
dvc add {{caminho/para/diretorio}}
```

- Adiciona recursivamente todos os arquivos em um dado diretório:

```
dvc add --recursive {{caminho/para/diretorio}}
```

- Adiciona um arquivo com o nome `.dvc` customizado:

```
dvc add --file {{nome_customizado.dvc}} {{caminho/para/arquivo}}
```

# dvc checkout

Registra a saída de arquivos e diretórios de dados vindos do cache.

Mais informações: <https://dvc.org/doc/command-reference/checkout>.

- Registra a saída de todos os arquivos e diretórios de dados:

```
dvc checkout
```

- Registra a saída da última versão de um alvo específico:

```
dvc checkout {{alvo}}
```

- Registra a saída de versão específica de um alvo de um commit/tag/branch  
Git:

```
git checkout {{hash_do_commit|tag|branch}} {{alvo}} && dvc  
checkout {{alvo}}
```

# dvc

Controle de Versão de Dados: como o **git** mas para dados.

Alguns subcomandos tal como **dvc commit** tem suas próprias documentações de uso.

Mais informações: <https://dvc.org/>.

- Checa a versão do DVC:

```
dvc --version
```

- Exibe a ajuda geral:

```
dvc --help
```

- Exibe ajuda para um subcomando específico:

```
dvc {{subcomando}} --help
```

- Executa um subcomando do DVC:

```
dvc {{subcomando}}
```

# ebook-convert

Pode ser usado para converter e-books entre os formatos comuns, como PDF, EPUB e MOBI.

Faz parte da biblioteca de ferramentas Calibre e-book.

Mais informações: <https://manual.calibre-ebook.com/generated/en/ebook-convert.html>.

- Converte um e-book em outro formato:

```
ebook-convert {{caminho/para/arquivo_entrada}}  
{{arquivo_saída}}
```

- Converte Markdown ou HTML em um e-book com ToC, título e autor:

```
ebook-convert {{caminho/para/arquivo_entrada}}  
{{arquivo_saída}} --level1-toc="//h:h1" --level2-toc="//h:h2"  
--level3-toc="//h:h3" --title={{título}} --authors={{autor}}
```



# echo

Imprime os argumentos passados.

Mais informações: <https://www.gnu.org/software/coreutils/echo>.

- Imprime uma mensagem de texto. Nota: aspas são opcionais:

```
echo "{{Olá Mundo}}"
```

- Imprime uma mensagem com variáveis de ambiente:

```
echo "{{Meu caminho é $PATH}}"
```

- Imprime uma mensagem sem adicionar uma nova linha no final:

```
echo -n "{{Olá Mundo}}"
```

- Adiciona uma mensagem no arquivo:

```
echo "{{Olá Mundo}}" >> {{arquivo.txt}}
```

- Habilita interpretação dos códigos de escape após barra invertida (caracteres especiais):

```
echo -e "{{Coluna 1\tColuna 2}}"
```

- Imprime o status de saída do último comando executado (Nota: no prompt de comando do Windows e no PowerShell, os comandos equivalentes são `echo %errorlevel%` e `$lastexitcode` respectivamente):

```
echo $?
```

# ect

Efficient Compression Tool.

Otimizador de arquivos escrito em C++. Suporta arquivos do tipo **.png**, **.jpg**, **.gzip** and **.zip**.

Mais informações: <https://github.com/fhanau/Efficient-Compression-Tool>.

- Comprime um arquivo:

```
ect {{caminho/para/arquivo.png}}
```

- Comprime um arquivo com level de compressão específico e multithreading (1=Mais rápido (pior), 9=Mais lento (Melhor). O padrão é 3):

```
ect -{{9}} --mt-deflate {{caminho/para/arquivo.zip}}
```

- Comprime todos os arquivos em um diretório recursivamente:

```
ect -recurse {{caminho/para/diretório}}
```

- Comprime um arquivo, mantendo o tempo de modificação original:

```
ect -keep {{caminho/para/arquivo.png}}
```

- Comprime um arquivo, removendo metadados:

```
ect -strip {{caminho/para/arquivo.png}}
```

# ed

O editor de texto original do Unix.

Veja também: **awk**, **sed**.

Mais informações: [https://www.gnu.org/software/ed/manual/ed\\_manual.html](https://www.gnu.org/software/ed/manual/ed_manual.html).

- Inicia uma sessão do editor interativo com um documento vazio:

```
ed
```

- Inicia uma sessão do editor interativo com um documento vazio e um prompt específico:

```
ed --prompt='> '
```

- Inicia uma sessão do editor interativo com erros compreensíveis para usuários:

```
ed --verbose
```

- Inicia uma sessão do editor interativo com um documento vazio e sem diagnósticos, contagens de bytes e prompt "!":

```
ed --quiet
```

- Inicia uma sessão do editor interativo sem mudança no status de saída quando o comando falha:

```
ed --loose-exit-status
```

- Edita um arquivo específico (mostra a contagem de bytes do arquivo carregado):

```
ed {{caminho/para/arquivo}}
```

- Substitui a string com um substituto específico em todas as linhas:

```
,s/{{regular_expression}}/{{substituto}}/g
```

# emacs

O editor extensível, personalizável, autodocumentável, com exibição em tempo real.

Veja também **emacsclient**.

Mais informações: <https://www.gnu.org/software/emacs>.

- Inicia o Emacs e abra um arquivo:

```
emacs {{caminho/para/arquivo}}
```

- Abre um arquivo em uma linha especificada:

```
emacs +{{numero_linha}} {{caminho/para/arquivo}}
```

- Inicia o Emacs em modo console (sem uma janela X):

```
emacs --no-window-system
```

- Inicia um servidor Emacs em segundo plano (acessível através do `emacsclient`):

```
emacs --daemon
```

- Para um servidor Emacs em funcionamento e todas as suas instâncias, pedindo confirmação em arquivos não salvos:

```
emacsclient --eval '(save-buffers-kill-emacs)'
```

- Salva um arquivo em Emacs:

```
Ctrl + X, Ctrl + S
```

- Deixa o Emacs:

```
Ctrl + X, Ctrl + C
```

# esbuild

Empacotador e minificador JavaScript construído para velocidade.

Mais informações: <https://esbuild.github.io/>.

- Empacota uma aplicação JavaScript e imprime para stdout:

```
esbuild --bundle {{caminho/para/arquivo.js}}
```

- Empacota uma aplicação JSX de stdin:

```
esbuild --bundle --outfile={{caminho/para/saída.js}} <
{{caminho/para/arquivo.jsx}}
```

- Empacota e reduz uma aplicação JSX com mapas de origem no modo production:

```
esbuild --bundle --define:
{{process.env.NODE_ENV=\"production\"}} --minify --sourcemap
{{caminho/para/arquivo.js}}
```

- Empacota uma aplicação JSX para uma lista de navegadores separados por vírgulas:

```
esbuild --bundle --minify --sourcemap --
target={{chrome58,firefox57,safari11,edge16}} {{caminho/para/
arquivo.jsx}}
```

- Empacota uma aplicação JavaScript para uma versão específica do node:

```
esbuild --bundle --platform={{node}} --target={{node12}}
{{caminho/para/arquivo.js}}
```

- Empacota uma aplicação JavaScript habilitando a sintaxe JSX em arquivos .js:

```
esbuild --bundle app.js --loader:{{.js=jsx}} {{caminho/para/
arquivo.js}}
```

- Empacota e serve uma aplicação JavaScript em um servidor HTTP:

```
esbuild --bundle --serve={{porta}} --outfile={{index.js}}
{{caminho/para/arquivo.js}}
```

- Empacota uma lista de arquivos em um diretório de saída:

```
esbuild --bundle --outdir={{caminho/para/diretório_de_saída}}
{{caminho/para/arquivo1 caminho/para/arquivo2 ...}}
```

# fast

Teste sua velocidade de download e upload utilizando fast.com.

Mais informações: <https://github.com/sindresorhus/fast-cli>.

- Mede a velocidade de download atual:

```
fast
```

- Mede a velocidade de upload atual além da velocidade de download:

```
fast --upload
```

- Exibe os resultados em uma única linha para reduzir espaçamento:

```
fast --single-line
```

# fastboot

Se comunica com dispositivos Android conectados quando iniciados no modo fastboot (o único lugar em que **adb** não funciona).

Mais informações: <https://cs.android.com/android/platform/superproject/+/main:system/core/fastboot>.

- Desbloqueia o bootloader:

```
fastboot oem unlock
```

- Bloqueia o bootloader novamente:

```
fastboot oem lock
```

- Reinicia o dispositivo no modo fastboot para o modo fastboot novamente:

```
fastboot reboot bootloader
```

- Flasheia uma imagem:

```
fastboot flash {{arquivo.img}}
```

- Flasheia uma imagem de recovery customizada:

```
fastboot flash recovery {{arquivo.img}}
```

- Exibe os dispositivos conectados:

```
fastboot devices
```

- Mostra todas as informações de um dispositivo:

```
fastboot getvar all
```

# fc-list

Exibe todas as fontes disponíveis no sistema.

Mais informações: <https://manned.org/fc-list>.

- Exibir as fontes instaladas correspondentes ao critério de busca:

```
fc-list | grep '{{critério_de_busca}}'
```



# fc

Abre o último comando executado em um editor de texto.

Mais informações: <https://manned.org/fc>.

- Abrir o último comando executado no editor de texto padrão do sistema:

```
fc
```

- Especificar o editor de texto que será utilizado ao executar o comando:

```
fc -e {'emacs'}
```

- Exibir um histórico dos últimos comandos executados:

```
fc -l
```

# fd

Uma alternativa para **find**.

Visa ser mais rápido e fácil de usar do que **find**.

Mais informações: <https://github.com/sharkdp/fd>.

- Encontra recursivamente arquivos que correspondam ao padrão fornecido no diretório atual:

```
fd {{padrão}}
```

- Encontra arquivos que começam com foo:

```
fd '^foo'
```

- Encontra arquivos com uma extensão específica:

```
fd --extension txt
```

- Encontra arquivos em um diretório específico:

```
fd '{{padrão}}' {{caminho/para/diretório}}
```

- Inclui arquivos ignorados e ocultos na pesquisa:

```
fd --hidden --no-ignore '{{padrão}}'
```

- Executa um comando em cada resultado de pesquisa retornado:

```
fd '{{padrão}}' --exec {{comando}}
```

# fdp

Renderiza uma imagem de um gráfico de rede **force-directed** a partir de um arquivo **graphviz**.

Layouts: **dot**, **neato**, **twopi**, **circo**, **fdp**, **sfdp**, **osage** & **patchwork**.

Mais informações: <https://graphviz.org/doc/info/command.html>.

- Renderiza uma imagem **png** com um nome de arquivo baseado no nome do arquivo de entrada e formato de saída (-O maiúsculo):

```
fdp -T png -O {{caminho/para/entrada.gv}}
```

- Renderiza uma imagem **svg** com o nome do arquivo de saída especificado (-o minúsculo):

```
fdp -T svg -o {{caminho/para/imagem.svg}} {{caminho/para/entrada.gv}}
```

- Renderiza a saída nos formatos:

```
fdp -T {{ps|pdf|svg|fig|png|gif|jpg|json|dot}} -O {{caminho/para/entrada.gv}}
```

- Renderiza uma imagem **gif** usando **stdin** e **stdout**:

```
echo "{{digraph {isso -> aquilo} }}" | fdp -T gif > {{caminho/para/imagem.gif}}
```

- Exibe ajuda:

```
fdp -?
```

# ffmpeg

Ferramenta de conversão de vídeo.

Mais informações: <https://ffmpeg.org>.

- Extrair o som de um vídeo e salvá-lo como MP3:

```
ffmpeg -i {{vídeo}} -vn {{som}}.mp3
```

- Converter quadros de um vídeo ou GIF para imagens numeradas individuais:

```
ffmpeg -i {{vídeo|gif}} {{quadro_%d.png}}
```

- Combinar imagens numeradas (quadro\_1.jpg, quadro\_2.jpg, etc) em um vídeo ou GIF:

```
ffmpeg -i {{quadro_%d.jpg}} -f image2 {{vídeo|gif}}
```

- Extrair um único quadro de um vídeo no tempo mm:ss e salvá-lo como uma imagem de resolução 128x128:

```
ffmpeg -ss {{mm:ss}} -i {{vídeo}} -frames 1 -s {{128x128}} -f image2 {{quadro.png}}
```

- Cortar um vídeo de um dado tempo inicial mm:ss até um tempo final mm2:ss2 (omite a opção -to para cortar o vídeo até o final):

```
ffmpeg -ss {{mm:ss}} -to {{mm2:ss2}} -i {{vídeo_entrada}} -codec copy {{vídeo_saída}}
```

- Converter um vídeo AVI para MP4. AAC Áudio @ 128kbit, h264 Vídeo @ CRF 23:

```
ffmpeg -i {{vídeo_entrada}}.avi -codec:audio aac -b:audio 128k -codec:video libx264 -crf 23 {{vídeo_saída}}.mp4
```

- Remuxar um vídeo MKV para MP4 sem recodificar áudio ou vídeo:

```
ffmpeg -i {{vídeo_entrada}}.mkv -codec copy {{vídeo_saída}}.mp4
```

- Converter vídeo MP4 para o codec VP9. Para a melhor qualidade, use um valor CRF (faixa recomendada 15-35) e -b:video DEVE ser 0:

```
ffmpeg -i {{vídeo_entrada}}.mp4 -codec:video libvpx-vp9 -crf {{30}} -b:video 0 -codec:audio libopus -vbr on -threads {{número_de_threads}} {{vídeo_saída}}.webm
```

# finger

Programa de pesquisa de informações do usuário.

Mais informações: <https://manned.org/finger>.

- Exibe informações sobre usuários conectados no momento:

```
finger
```

- Exibe informações sobre um usuário específico:

```
finger {{nome_de_usuario}}
```

- Exibe o nome de login do usuário, nome real, nome do terminal e outras informações:

```
finger -s
```

- Produz o formato de saída de várias linhas exibindo as mesmas informações que `-s`, bem como o diretório pessoal do usuário, número de telefone residencial, shell de login, status de correio, etc.:

```
finger -l
```

- Impede a correspondência com os nomes de usuário e usa apenas nomes de login:

```
finger -m
```

# firefox

Um browser livre e de código aberto.

Mais informações: [https://developer.mozilla.org/en-US/docs/Mozilla/Command Line Options](https://developer.mozilla.org/en-US/docs/Mozilla/Command_Line_Options).

- Inicie o Firefox e abra uma página web:

```
firefox {{https://www.duckduckgo.com}}
```

- Abra uma nova janela:

```
firefox --new-window {{https://www.duckduckgo.com}}
```

- Abra uma janela privativa (incognito):

```
firefox --private-window
```

- Pesquise por "wikipedia" usando a engine de busca padrão:

```
firefox --search "{{wikipedia}}"
```

- Inicie o Firefox no modo seguro, com todas as extensões desativadas:

```
firefox --safe-mode
```

- Tire uma screenshot de uma página web no modo headless:

```
firefox --headless --screenshot {{caminho/para/arquivo_de_saida.png}} {{https://exemplo.com/}}
```

- Use um perfil específico para permitir que múltiplas instâncias separadas do Firefox rodem ao mesmo tempo:

```
firefox --profile {{caminho/para/diretório}} {{https://example.com/}}
```

- Configure o Firefox como o navegador padrão:

```
firefox --setDefaultBrowser
```

# flips

Cria e aplica patches em arquivos IPS e BPS.

Mais informações: <https://github.com/Alcaro/Flips>.

- Abre Flips para criar e aplicar um patch:

```
flips
```

- Aplica um patch criando um novo arquivo ROM:

```
flips --apply {{patch.bps}} {{rom.smc}} {{hack.smc}}
```

- Cria um patch a partir de duas ROMs:

```
flips --create {{rom.smc}} {{hack.smc}} {{patch.bps}}
```

# flutter

SDK livre e open source do Google para desenvolvimento de aplicativos mobile cross-platform.

Mais informações: <https://github.com/flutter/flutter/wiki/The-flutter-tool>.

- Mostra ajuda sobre algum comando específico:

```
flutter help {{comando}}
```

- Verifica se todas as ferramentas externas necessárias estão instaladas:

```
flutter doctor
```

- Lista ou muda o channel do Flutter:

```
flutter channel {{stable|beta|dev|master}}
```

- Executa o projeto Flutter em todos os emuladores ativos ou devices conectados:

```
flutter run -d all
```

- Instala todas as dependências definidas no `pubspec.yaml`:

```
flutter pub get
```

- Executa todos os testes no terminal a partir da raiz do projeto:

```
flutter test {{test/example_test.dart}}
```

- Buildar APK de release direcionado aos mais modernos smartphones:

```
flutter build apk --target-platform {{android-arm}},  
{{android-arm64}}
```



# fmt

Reformata um arquivo de texto juntando seus parágrafos e limitando a largura da linha a um determinado número de caracteres (75 por padrão).

Mais informações: <https://www.gnu.org/software/coreutils/fmt>.

- Reformata um arquivo:

```
fmt {{caminho/para/arquivo}}
```

- Reformata um arquivo produzindo linhas de saída de (no máximo) *n* caracteres:

```
fmt -w {{n}} {{caminho/para/arquivo}}
```

- Reformata um arquivo sem unir linhas menores do que a largura fornecida:

```
fmt -s {{caminho/para/arquivo}}
```

- Reformata um arquivo com espaçamento uniforme (1 espaço entre palavras e 2 espaços entre parágrafos):

```
fmt -u {{caminho/para/arquivo}}
```

# fnm

Gerenciador de versão Node.js rápido.

Instale, Desinstale ou alterne entre as versões do Node.js.

Mais informações: <https://github.com/Schniz/fnm>.

- Instala uma versão específica do Node.js:

```
fnm install {{versão_do_node}}
```

- Lista todas as versões disponíveis do Node.js e destaca a padrão:

```
fnm ls
```

- Usa uma versão específica do Node.js no shell atual:

```
fnm use {{versão_do_node}}
```

- Define a versão padrão do Node.js:

```
fnm default {{versão_do_node}}
```

- Desinstala uma determinada versão do Node.js:

```
fnm uninstall {{versão_do_node}}
```

# fossil-ci

Este comando é um pseudônimo de **fossil-commit**.

Mais informações: <https://fossil-scm.org/home/help/commit>.

- Ver documentação sobre o comando original:

```
tldr fossil-commit
```

# fossil-delete

Este comando é um pseudônimo de **fossil rm**.

Mais informações: <https://fossil-scm.org/home/help/delete>.

- Ver documentação sobre o comando original:

```
tldr fossil rm
```

# fossil-forget

Este comando é um pseudônimo de **fossil rm**.

Mais informações: <https://fossil-scm.org/home/help/forget>.

- Ver documentação sobre o comando original:

```
tldr fossil rm
```

# fossil-new

Este comando é um pseudônimo de **fossil-init**.

Mais informações: <https://fossil-scm.org/home/help/new>.

- Ver documentação sobre o comando original:

```
tldr fossil-init
```

# ftp

Ferramentas para interagir com um servidor via Protocolo de Transferência de Arquivos.

Mais informações: <https://manned.org/ftp>.

- Conecta-se a um servidor FTP:

```
ftp {{ftp.exemplo.com}}
```

- Alterna para o modo de transferência binária (gráficos, arquivos compactados, etc):

```
binary
```

- Transfere vários arquivos sem solicitar confirmação em cada arquivo:

```
prompt off
```

- Baixa vários arquivos (expressão glob):

```
mget {{*.png}}
```

- Carrega vários arquivos (expressão glob):

```
mput {{*.zip}}
```

- Exclui vários arquivos no servidor remoto:

```
mdelete {{*.txt}}
```

- Renomeia um arquivo no servidor remoto:

```
rename {{nome_do_arquivo_original}} {{nome_do_novo_arquivo}}
```

# gplusplus

Compila arquivos de código fonte C++.

Parte do GCC (GNU Compiler Collection).

Mais informações: <https://gcc.gnu.org>.

- Compilar um arquivo de código fonte para um binário executável:

```
g++ {{caminho/para/fonte.cpp}} -o {{caminho/para/
executável_de_saída}}
```

- Exibe avisos comuns:

```
g++ {{caminho/para/fonte.cpp}} -Wall -o {{caminho/para/
executável_de_saída}}
```

- Escolha um padrão de linguagem para o qual compilar (C++98/C++11/C++14/C++17):

```
g++ {{caminho/para/fonte.cpp}} -std={{c++98|c++11|c++14|c+
+17}} -o {{caminho/para/executável_de_saída}}
```

- Inclui bibliotecas localizadas em um caminho diferente do arquivo de código fonte:

```
g++ {{caminho/para/fonte.cpp}} -o {{caminho/para/
executável_de_saída}} -I{{caminho/para/cabeçalho}} -
L{{caminho/para/biblioteca}} -l{{nome_da_biblioteca}}
```

- Compila e vincula múltiplos arquivos de código fonte em um binário executável:

```
g++ -c {{caminho/para/fonte_1.cpp caminho/para/fonte_2.cpp
...}} && g++ -o {{caminho/para/executável_de_saída}}
{{caminho/para/fonte_1.o caminho/para/fonte_2.o ...}}
```

- Exibe versão:

```
g++ --version
```



# gatsby

Gerador de site estático para React.

Mais informações: <https://gatsbyjs.org>.

- Cria um novo site:

```
gatsby new {{nome_do_site}}
```

- Cria um novo site com um 'starter' do Gatsby:

```
gatsby new {{nome_do_site}}  
{{url_do_repositório_github_do_starter}}
```

- Inicia um servidor de desenvolvimento local de atualização em tempo real:

```
gatsby develop
```

- Executa uma build de produção e gera HTML estático:

```
gatsby build
```

- Inicia um servidor local que atende a build de produção:

```
gatsby serve
```

# gcal

Exibe o calendário.

Mais informações: <https://www.gnu.org/software/gcal>.

- Exibe o calendário para o mês atual:

```
gcal
```

- Exibe o calendário para o mês de Fevereiro do ano de 2010:

```
gcal {{2}} {{2010}}
```

- Fornece folha de calendário com números da semana:

```
gcal --with-week-number
```

- Altera o dia da semana de início para o 1º dia da semana (segunda-feira):

```
gcal --starting-day={{1}}
```

- Exibe o mês anterior, atual e seguinte em torno de hoje:

```
gcal .
```

# gcalcli

Ferramenta de linha de comando para interagir com o Google Agenda.

Solicita autorização da API do Google na primeira inicialização.

Mais informações: <https://github.com/insanum/gcalcli>.

- Lista seus eventos para todos os seus calendários nos próximos 7 dias:

```
gcalcli agenda
```

- Mostra eventos começando em ou entre datas específicas (também recebe datas relativas, por exemplo, "amanhã"):

```
gcalcli agenda {{mm/dd}} [{{mm/dd}}]
```

- Lista eventos de um calendário específico:

```
gcalcli --calendar {{nome_do_calendário}} agenda
```

- Exibe um calendário ASCII de eventos por semana:

```
gcalcli calw
```

- Exibe um calendário ASCII de eventos para um mês:

```
gcalcli calm
```

- Adiciona um evento rapidamente ao seu calendário:

```
gcalcli --calendar {{nome_do_calendário}} quick "{{mm/dd}}  
{{HH:MM}} {{nome_do_evento}}"
```

- Adiciona um evento ao calendário. Dispara prompt interativo:

```
gcalcli --calendar "{{nome_do_calendário}}" add
```

# gcc

Compilador de arquivos de código fonte C e C++, efetuando também as fases de pré-processamento, assembling e linking.

Mais informações: <https://gcc.gnu.org>.

- Compilar múltiplos arquivos de código fonte, produzindo um arquivo executável:

```
gcc {{arquivo_fonte1.c}} {{arquivo_fonte2.c}} --output  
{{arquivo_executável}}
```

- Habilitar avisos durante a compilação:

```
gcc {{arquivo_fonte.c}} -Wall -Og --output  
{{arquivo_executável}}
```

- Incluir bibliotecas de um local diferente:

```
gcc {{arquivo_fonte.c}} --output {{arquivo_executável}} -  
I{{caminho/para/header}} -L{{caminho/para/biblioteca}} -  
l{{nome_biblioteca}}
```

- Compilar o código fonte para instruções Assembler:

```
gcc -S {{arquivo_fonte.c}}
```

- Compilar o código fonte sem efetuar a fase de linking:

```
gcc -c {{arquivo_fonte.c}}
```

# gdb

O depurador GNU.

Mais informações: <https://www.gnu.org/software/gdb>.

- Depurar um executável:

```
gdb {{executável}}
```

- Vincular um processo ao gdb:

```
gdb -p {{PID}}
```

- Depurar usando um arquivo de "core dump":

```
gdb -c {{core}} {{executável}}
```

- Executa um dado comando do gdb ao iniciar:

```
gdb -ex "{{comandos}}" {{executável}}
```

- Inicia o gdb passando argumentos para o executável:

```
gdb --args {{executável}} {{argumento1}} {{argumento2}}
```

# gdu

Analizador de uso de disco com interface de console.

Mais informações: <https://github.com/dundee/gdu>.

- Exibe interativamente o uso de disco do diretório atual:

```
gdu
```

- Exibe interativamente o uso de disco de um determinado diretório:

```
gdu {{caminho/para/diretório}}
```

- Exibe interativamente o uso de disco de todos os discos montados:

```
gdu --show-disks
```

- Exibe interativamente o uso de disco do diretório atual, mas ignora alguns subdiretórios:

```
gdu --ignore-dirs {{caminho/para/diretório1,caminho/para/diretório2,...}}
```

- Ignora caminhos por expressão regular:

```
gdu --ignore-dirs-pattern '{{.*[abc]+}}'
```

- Ignora diretórios ocultos:

```
gdu --no-hidden
```

- Imprime apenas o resultado, não entra no modo interativo:

```
gdu --non-interactive {{caminho/para/diretório}}
```

- Não mostra o progresso no modo não interativo (útil em scripts):

```
gdu --no-progress {{caminho/para/diretório}}
```

# gh-cs

Este comando é um pseudônimo de **gh-codespace**.

Mais informações: [https://cli.github.com/manual/gh\\_codespace](https://cli.github.com/manual/gh_codespace).

- Ver documentação sobre o comando original:

```
tldr gh-codespace
```

# git add

Adiciona arquivos modificados na área de preparação.

Mais informações: <https://git-scm.com/docs/git-add>.

- Adiciona um arquivo na área de preparação:

```
git add {{caminho/do/arquivo}}
```

- Adiciona todos arquivos (rastreados ou não):

```
git add -A
```

- Adiciona apenas arquivos rastreados:

```
git add -u
```

- Adiciona arquivos ignorados:

```
git add -f
```

- Interativamente adiciona partes dos arquivo:

```
git add -p
```

- Interativamente adiciona partes de um dado arquivo:

```
git add -p {{caminho/para/arquivo}}
```

- Interativamente adiciona arquivos ou partes modificadas:

```
git add -i
```



# git alias

Cria comandos curtos para comandos Git.

Parte do **git-extras**.

Mais informações: <https://github.com/tj/git-extras/blob/master/Commands.md#git-alias>.

- Lista todos os comandos curtos:

```
git alias
```

- Cria um novo comando curto com o nome:

```
git alias "{{nome}}" "{{comando}}"
```

- Procura por um comando curto existente:

```
git alias ^{{nome}}
```

# git branch

Comando principal do Git para trabalhar com branches.

Mais informações: <https://git-scm.com/docs/git-branch>.

- Lista todas as branches (locais e remotas; a branch atual é destacada por \*):

```
git branch --all
```

- Lista quais branches incluem um commit específico do Git em seu histórico:

```
git branch --all --contains {{hash_do_commit}}
```

- Mostra o nome da branch atual:

```
git branch --show-current
```

- Cria uma nova branch baseada no commit atual:

```
git branch {{nome_da_branch}}
```

- Cria uma nova branch baseada em um commit específico:

```
git branch {{nome_da_branch}} {{hash_do_commit}}
```

- Renomeia uma branch (não precisa fazer checkout para isso):

```
git branch -m {{antigo_nome_da_branch}}  
{{novo_nome_da_branch}}
```

- Exclui a branch local (não precisa fazer checkout para isso):

```
git branch -d {{nome_da_branch}}
```

- Exclui uma branch remota:

```
git push {{nome_remoto}} --delete {{nome_da_branch_remota}}
```

# git checkout

Faz checkout de uma branch ou caminhos para uma árvore de trabalho.

Mais informações: <https://git-scm.com/docs/git-checkout>.

- Cria e muda para uma nova branch:

```
git checkout -b {{nome_da_branch}}
```

- Cria e muda para uma nova branch com base em uma referência específica (branch, remoto/branch, etiqueta são exemplos de referências válidas):

```
git checkout -b {{nome_da_branch}} {{referência}}
```

- Muda para uma branch local existente:

```
git checkout {{nome_da_branch}}
```

- Muda para uma branch previamente verificada:

```
git checkout -
```

- Muda para uma branch remota existente:

```
git checkout --track {{nome_remoto}}/{{nome_da_branch}}
```

- Descarta todas as alterações não preparadas no diretório atual (consulte `git reset` para mais comandos do tipo desfazer):

```
git checkout .
```

- Descarta alterações não preparadas em um determinado arquivo:

```
git checkout {{caminho/para/arquivo}}
```

- Substitui um arquivo no diretório atual com a versão com commit em uma determinada branch:

```
git checkout {{nome_da_branch}} -- {{caminho/para/arquivo}}
```

# git clone

Clona um repositório existente.

Mais informações: <https://git-scm.com/docs/git-clone>.

- Clona um repositório existente em um novo diretório (o diretório padrão é o nome do repositório):

```
git clone {{local_do_repositório_remoto}} {{caminho/para/diretório}}
```

- Clona um repositório existente e seus submódulos:

```
git clone --recursive {{local_do_repositório_remoto}}
```

- Clona somente o diretório `.git` de um repositório existente:

```
git clone --no-checkout {{local_do_repositório_remoto}}
```

- Clona um repositório local:

```
git clone --local {{caminho/para/repositório/local}}
```

- Clona de forma silenciosa:

```
git clone --quiet {{local_do_repositório_remoto}}
```

- Clona um repositório existente buscando somente os 10 commits mais recentes na branch padrão (útil para salvar tempo):

```
git clone --depth {{10}} {{local_do_repositório_remoto}}
```

- Clona um repositório existente buscando somente uma branch específica:

```
git clone --branch {{nome}} --single-branch  
{{local_do_repositório_remoto}}
```

- Clona um repositório existente usando um comando SSH específico:

```
git clone --config core.sshCommand="{{ssh -i caminho/para/chave_ssh_privada}}" {{local_do_repositório_remoto}}
```

# git commit

Faz um commit dos arquivos no repositório.

Mais informações: <https://git-scm.com/docs/git-commit>.

- Faz um commit com os arquivos preparados no repositório com uma mensagem:

```
git commit --message "{{mensagem}}"
```

- Faz um commit com os arquivos preparados com uma mensagem lida de um arquivo:

```
git commit --file {{caminho/para/
arquivo_de_mensagem_do_commit}}
```

- Prepara automaticamente todos os arquivos modificados e excluídos e faz o commit com uma mensagem:

```
git commit --all --message "{{mensagem}}"
```

- Faz um commit com os arquivos preparados e assina-os com a chave GPG especificada (ou a definida no arquivo de configuração se nenhum argumento for especificado):

```
git commit --gpg-sign {{id_da_chave}} --message
 "{{mensagem}}"
```

- Atualiza o último commit adicionando as alterações atualmente preparadas, alterando o hash do commit:

```
git commit --amend
```

- Faz um commit apenas de arquivos específicos (já preparados):

```
git commit {{caminho/para/arquivo1}} {{caminho/para/
arquivo2}}
```

- Cria um commit, mesmo se não haja arquivos preparados:

```
git commit --message "{{mensagem}}" --allow-empty
```

# git config

Gerencia configurações customizadas para repositórios Git.

Estas configurações podem ser locais (para o repositório corrente) ou globais (para o usuário atual).

Mais informações: <https://git-scm.com/docs/git-config>.

- Lista somente configurações locais (armazenadas no `.git/config` do repositório corrente):

```
git config --list --local
```

- Lista somente configurações globais (armazenadas no `~/.gitconfig` por padrão ou no `$XDG_CONFIG_HOME/git/config` se tal arquivo existe):

```
git config --list --global
```

- Lista somente configurações do sistema (armazenadas no `/etc/gitconfig`), e exibe o local do arquivo:

```
git config --list --system --show-origin
```

- Obtém o valor de uma dada variável de configuração:

```
git config alias.unstage
```

- Armazena o valor global de uma dada variável de configuração:

```
git config --global alias.unstage "reset HEAD --"
```

- Reverte o valor global de uma dada variável de configuração para seu valor padrão:

```
git config --global --unset alias.unstage
```

- Edita a configuração Git do repositório corrente usando o editor padrão:

```
git config --edit
```

- Edita a configuração global do Git usando o editor padrão:

```
git config --global --edit
```

# git diff

Mostra alterações nos arquivos rastreados.

Mais informações: <https://git-scm.com/docs/git-diff>.

- Mostra as alterações não preparadas e sem commit:

```
git diff
```

- Mostra todas as alterações sem commit (incluindo as preparadas):

```
git diff HEAD
```

- Mostra apenas as alterações preparadas (adicionadas, mas ainda sem commit):

```
git diff --staged
```

- Mostra as alterações de todos os commits desde uma determinada data/hora (uma expressão de data, por exemplo, "1 week 2 days" ou uma data ISO):

```
git diff 'HEAD@{3 months|weeks|days|hours|seconds ago}'
```

- Mostra apenas os nomes dos arquivos alterados desde um determinado commit:

```
git diff --name-only {{commit}}
```

- Emite um resumo das criações de arquivos, renomeações e alterações de modo desde um determinado commit:

```
git diff --summary {{commit}}
```

- Compara um único arquivo entre duas branches ou commits:

```
git diff {{branch_1}}..{{branch_2}} [--] {{caminho/para/arquivo}}
```

- Compara diferentes arquivos da branch atual com outra branch:

```
git diff {{branch}}:{{caminho/para/arquivo2}} {{caminho/para/arquivo}}
```

# git gc

Otimiza o repositório local limpando os arquivos desnecessários.

Mais informações: <https://git-scm.com/docs/git-gc>.

- Otimiza o repositório:

```
git gc
```

- Otimiza de forma mais agressiva, demora mais tempo:

```
git gc --aggressive
```

- Não remove objetos perdidos (por default é removido):

```
git gc --no-prune
```

- Não exibe a saída:

```
git gc --quiet
```

- Exibe toda a ajuda:

```
git gc --help
```



# git init

Inicializa um novo repositório Git local.

Mais informações: <https://git-scm.com/docs/git-init>.

- Inicializa um novo repositório local:

```
git init
```

- Inicializa um repositório com o nome especificado para a branch inicial:

```
git init --initial-branch={{nome_da_branch}}
```

- Inicializa um repositório usando SHA256 para os hashes de objeto (requer Git versão 2.29+):

```
git init --object-format={{sha256}}
```

- Inicializa um repositório barebones, adequado para usar como um remoto via ssh:

```
git init --bare
```

# git log

Mostra um histórico de commits.

Mais informações: <https://git-scm.com/docs/git-log>.

- Mostra a sequência de commits a partir do atual, em ordem cronológica reverse do repositório Git no diretório de trabalho atual:

```
git log
```

- Mostra o histórico de um arquivo ou diretório determinado, incluindo diferenças:

```
git log -p {{caminho/para/arquivo_ou_diretório}}
```

- Mostra uma visão geral do(s) arquivo(s) alterado(s) em cada commit:

```
git log --stat
```

- Mostra um grafo dos commits no branch atual usando apenas a primeira linha de cada mensagem de commit:

```
git log --oneline --graph
```

- Mostra um grafo de todos os commits, etiquetas e branches em todo o repositório:

```
git log --oneline --decorate --all --graph
```

- Mostra apenas os commits cujas mensagem incluem uma determinada cadeia de caracteres (sem distinção entre maiúsculas e minúsculas):

```
git log -i --grep {{cadeia_de_caracteres_para_pesquisa}}
```

- Mostra os últimos N commits de um determinado autor:

```
git log -n {{número}} --author={{autor}}
```

- Mostra os commits entre duas datas(aaaa-mm-dd):

```
git log --before="{{2017-01-29}}" --after="{{2017-01-17}}"
```

# git mv

Move ou renomeia arquivos e atualiza o index do Git.

Mais informações: <https://git-scm.com/docs/git-mv>.

- Move arquivos dentro de um repositório e adiciona no próximo commit:

```
git mv {{caminho/para/arquivo}} {{novo/caminho}}
```

- Renomeia um arquivo e adiciona a renomeação no próximo commit:

```
git mv {{nome_do_arquivo}} {{novo_nome}}
```

- Sobrescreve o arquivo no caminho alvo se ele já existir:

```
git mv --force {{arquivo}} {{alvo}}
```

# git pr

Traz o código via checkout dos pull requests do GitHub localmente.

Parte do **git-extras**.

Mais informações: <https://github.com/tj/git-extras/blob/master/Commands.md#git-pr>.

- Traz o código específico de um pull request:

```
git pr {{numero_do_pr}}
```

- Traz o código de um pull request para um remoto específico:

```
git pr {{numero_do_pr}} {{remoto}}
```

- Traz o código de um pull request da sua URL:

```
git pr {{url}}
```

- Limpa pull requests antigos:

```
git pr clean
```

# git pull

Obtém branch de um repositório remoto e mescla-a ao repositório local.

Mais informações: <https://git-scm.com/docs/git-pull>.

- Baixa as alterações do repositório remoto padrão e mescla-as:

```
git pull
```

- Baixa as alterações do repositório remoto padrão e usa o avanço rápido:

```
git pull --rebase
```

- Baixa as alterações de um determinado repositório remoto e branch, então, mescla-as no HEAD:

```
git pull {{nome_remoto}} {{branch}}
```

# git push

Envia commits para um repositório remoto.

Mais informações: <https://git-scm.com/docs/git-push>.

- Envia alterações locais na branch atual para sua contraparte remota padrão:

```
git push
```

- Envia alterações de uma branch local específica para sua contraparte remota:

```
git push {{nome_remoto}} {{branch_local}}
```

- Envia alterações de uma branch local específica para sua contraparte remota, e define a branch remota como o destino push/pull padrão da branch local:

```
git push -u {{nome_remoto}} {{branch_local}}
```

- Envia alterações de uma branch local específica para uma branch remota específica:

```
git push {{nome_remoto}} {{branch_local}}:{{branch_remota}}
```

- Envia alterações em todas as branches locais para suas contrapartes em um determinado repositório remoto:

```
git push --all {{nome_remoto}}
```

- Exclui uma branch em um repositório remoto:

```
git push {{nome_remoto}} --delete {{branch_remota}}
```

- Remove branches remotas que não tenham uma contraparte local:

```
git push --prune {{nome_remoto}}
```

- Publica etiquetas que ainda não estão no repositório remoto:

```
git push --tags
```

# git rebase

Reaplica os commits de uma branch sobre outra branch.

Comumente usado para "mover" uma branch inteira para outra base, criando cópias dos commits na nova localização.

Mais informações: <https://git-scm.com/docs/git-rebase>.

- Faz um rebase na branch atual sobre outra branch especificada:

```
git rebase {{nova_branch_base}}
```

- Inicia um rebase interativo, que permite os commits serem reordenados, omitidos, combinados ou modificados:

```
git rebase -i {{branch_base_alvo_ou_hash_do_commit}}
```

- Continua um rebase que foi interrompido por uma falha de mesclagem, após a edição de arquivos conflitantes:

```
git rebase --continue
```

- Continua um rebase que foi pausado devido a conflitos de mesclagem, ignorando o commit conflitante:

```
git rebase --skip
```

- Aborta um rebase em andamento (por exemplo, se ele foi interrompido por um conflito de mesclagem):

```
git rebase --abort
```

- Move parte da branch atual para uma nova base, fornecendo a base antiga para começar:

```
git rebase --onto {{base_nova}} {{base_antiga}}
```

- Reaplica os últimos 5 commits no local, parando para permitir que eles sejam reordenados, omitidos, combinados ou modificados:

```
git rebase -i {{HEAD~5}}
```

- Resolve automaticamente quaisquer conflitos favorecendo a versão da branch de trabalho (a palavra-chave `theirs` tem significado invertido nesse caso):

```
git rebase -X theirs {{nome_da_branch}}
```



# git remote

Gerencia repositórios monitorados ("remotes").

Mais informações: <https://git-scm.com/docs/git-remote>.

- Mostre uma lista de remotes existentes, seus nomes e URL:

```
git remote -v
```

- Mostra informação de um remote específico:

```
git remote show {{nome_do_remote}}
```

- Adiciona um remote:

```
git remote add {{nome_do_remote}} {{url_do_remote}}
```

- Muda a URL de um remote (use - -add para manter a URL existente):

```
git remote set-url {{nome_do_remote}} {{nova_url}}
```

- Remove um remote:

```
git remote remove {{nome_do_remote}}
```

- Renomeia um remote:

```
git remote rename {{nome_antigo}} {{novo_nome}}
```

# git reset

Desfaz os commits ou as alterações não preparadas, redefinindo o Git HEAD atual para o estado especificado.

Se um caminho é passado, funcionará como "não preparado"; se um hash de commit ou uma branch é passado, funcionará como "sem commit".

Mais informações: <https://git-scm.com/docs/git-reset>.

- Remove tudo da preparação:

```
git reset
```

- Remove arquivo(s) específico(s) da preparação:

```
git reset {{caminho/para/arquivo1 caminho/para/arquivo2 ...}}
```

- Interativamente remove partes de um arquivo da preparação:

```
git reset --patch {{caminho/para/arquivo}}
```

- Desfaz o último commit, mantendo suas alterações (e quaisquer outras alteração não confirmadas) no sistema de arquivos:

```
git reset HEAD~
```

- Desfaz os últimos dois commits, adicionando suas alterações na área de preparação, isso é, preparando-os para o commit:

```
git reset --soft HEAD~2
```

- Descarta quaisquer alterações sem commit, preparadas ou não (para apenas alterações não preparadas, use o `git checkout`):

```
git reset --hard
```

- Redefine o repositório para um determinado commit, descartando as alterações com commit, preparadas e sem commit desde então:

```
git reset --hard {{commit}}
```

# git rm

Remove arquivos do índice do repositório e do arquivo local do sistema.

Mais informações: <https://git-scm.com/docs/git-rm>.

- Remove arquivo do índice do repositório e do arquivo local do sistema:

```
git rm {{arquivo}}
```

- Remove um diretório:

```
git rm -r {{diretório}}
```

- Remove arquivo do índice do repositório mas mantém o mesmo localmente:

```
git rm --cached {{arquivo}}
```

# git status

Mostra as alterações nos arquivos em um repositório Git.

Lista os arquivos alterados, adicionados e excluídos em comparação com o atual commit verificado.

Mais informações: <https://git-scm.com/docs/git-status>.

- Mostra arquivos alterados que ainda não foram adicionados para commit:

```
git status
```

- Fornece a saída em formato curto:

```
git status -s
```

- Não mostra arquivos não rastreados na saída:

```
git status --untracked-files=no
```

- Mostra a saída em formato curto junto com as informações da branch:

```
git status -sb
```

# git tag

Cria, exibe, exclui ou verifica tags.

Uma tag é uma referência estática para um commit específico.

Mais informações: <https://git-scm.com/docs/git-tag>.

- Exibe todas as tags:

```
git tag
```

- Crie uma tag com o nome fornecido apontando para o commit atual:

```
git tag {{nome_da_tag}}
```

- Crie uma tag com o nome fornecido apontando para um determinado commit:

```
git tag {{nome_da_tag}} {{commit}}
```

- Cria uma tag anotada com a mensagem fornecida:

```
git tag {{nome_da_tag}} -m {{mensagem_da_tag}}
```

- Exclui a tag com o nome fornecido:

```
git tag -d {{nome_da_tag}}
```

- Obtenha tags atualizadas do upstream:

```
git fetch --tags
```

- Liste todas as tags cujos ancestrais incluem um determinado commit:

```
git tag --contains {{commit}}
```

# git

Sistema de versionamento distribuído.

Alguns subcomandos como **git commit** tem sua própria documentação de uso.

Mais informações: <https://git-scm.com/>.

- Verifique a versão do Git:

```
git --version
```

- Mostre ajuda geral:

```
git --help
```

- Mostre ajuda de um subcomando do Git (como `commit`, `log`, etc.):

```
git help {{subcomando}}
```

- Execute um subcomando Git:

```
git {{subcomando}}
```

- Execute um subcomando Git no caminho raiz de um repositório específico:

```
git -C {{caminho/para/repo}} {{subcomando}}
```

- Execute um subcomando Git com uma dada configuração:

```
git -c '{{config.chave}}={{valor}}' {{subcomando}}
```

# gnmic-sub

Este comando é um pseudônimo de **gnmic subscribe**.

Mais informações: <https://gnmic.kmrd.dev/cmd/subscribe>.

- Ver documentação sobre o comando original:

```
tldr gnmic subscribe
```

# google-chrome

Este comando é um pseudônimo de **chromium**.

Mais informações: <https://chrome.google.com>.

- Ver documentação sobre o comando original:

```
tldr chromium
```



# gpg

GNU Privacy Guard.

Veja **gpg2** para o GNU Privacy Guard 2. A maioria dos sistemas operacionais criam link simbólico entre **gpg** e **gpg2**.

Mais informações: <https://gnupg.org>.

- Cria uma chave GPG pública e privada interativamente:

```
gpg --full-generate-key
```

- Assina doc.txt sem criptografia (cria um arquivo de saída doc.txt.asc):

```
gpg --clearsign {{doc.txt}}
```

- Criptografa e assina doc.txt para alice@example.com e bob@example.com (cria um arquivo de saída doc.txt.gpg):

```
gpg --encrypt --sign --recipient {{alice@example.com}} --recipient {{bob@example.com}} {{doc.txt}}
```

- Criptografa doc.txt apenas com uma senha simétrica (cria um arquivo de saída doc.txt.gpg):

```
gpg --symmetric {{doc.txt}}
```

- Descriptografa doc.txt.gpg (envia saída para stdout):

```
gpg --decrypt {{doc.txt.gpg}}
```

- Importa uma chave pública:

```
gpg --import {{public.gpg}}
```

- Exporta a chave pública da alice@example.com (envia saída para stdout):

```
gpg --export --armor {{alice@example.com}}
```

- Exporta chave privada da alice@example.com (envia saída para stdout):

```
gpg --export-secret-keys --armor {{alice@example.com}}
```

# grep

Acha padrões em arquivos usando expressões regulares.

Mais informações: <https://www.gnu.org/software/grep/manual/grep.html>.

- Pesquisa por um padrão em um arquivo:

```
grep "{{padrão_pesquisado}}" {{caminho/para/arquivo}}
```

- Pesquisa por uma string exata (desabilita expressões regulares):

```
grep --fixed-strings "{{string_exata}}" {{caminho/para/arquivo}}
```

- Pesquisa por um padrão em todos os arquivos recursivamente em um diretório, mostrando o número das linhas das correspondências, ignorando arquivos binários:

```
grep --recursive --line-number --binary-files={{without-match}} "{{padrão_pesquisado}}" {{caminho/para/diretório}}
```

- Usa expressões regulares estendidas (suporta ?, +, {}, () and |), no modo insensível a maiúsculas e minúsculas:

```
grep --extended-regexp --ignore-case "{{padrão_pesquisado}}" {{caminho/para/arquivo}}
```

- Imprime 3 linhas de contexto em volta, antes ou depois de cada correspondência:

```
grep --{{context|before-context|after-context}}={{3}}  
"{{padrão_pesquisado}}" {{caminho/para/arquivo}}
```

- Imprime o nome do arquivo e o número da linha para cada correspondência:

```
grep --with-filename --line-number "{{padrão_pesquisado}}"  
{{caminho/para/arquivo}}
```

- Pesquisa por linhas que correspondem a um padrão, imprimindo apenas o texto correspondido:

```
grep --only-matching "{{padrão_pesquisado}}" {{caminho/para/arquivo}}
```

- Pesquisa `stdin` para linhas que não correspondem a um padrão:

```
cat {{caminho/para/arquivo}} | grep --invert-match  
"{{padrão_pesquisado}}"
```

# gzip

Compacta/descompacta arquivos com compressão gzip (LZ77).

Mais informações: <https://www.gnu.org/software/gzip/manual/gzip.html>.

- Compacta um arquivo, substituindo-o por uma versão compactada gzip:

```
gzip {{arquivo.ext}}
```

- Descompacta um arquivo, substituindo-o pela versão não compactada original:

```
gzip -d {{arquivo.ext}}.gz
```

- Compacta um arquivo, mantendo o arquivo original:

```
gzip --keep {{arquivo.ext}}
```

- Compacta um arquivo definindo o nome do arquivo de saída:

```
gzip -c {{arquivo.ext}} > {{arquivo_compactado.ext.gz}}
```

- Descompacta um arquivo gzip definindo o nome do arquivo de saída:

```
gzip -c -d {{arquivo.ext}}.gz > {{arquivo_descompactado.ext}}
```

- Especifica o nível de compactação. 1=mais rápido (pior), 9=mais lento 9melhor, o nível padrão é 6:

```
gzip -9 -c {{arquivo.ext}} > {{arquivo_compactado.ext.gz}}
```

# helm

Helm é um gerenciador de pacotes para Kubernetes.

Mais informações: <https://helm.sh/>.

- Cria um chart do helm:

```
helm create {{nome_do_chart}}
```

- Adiciona um novo repositório helm:

```
helm repo add {{nome_do_repositório}}
```

- Lista os repositórios helm:

```
helm repo list
```

- Atualiza os repositórios helm:

```
helm repo update
```

- Remova um repositório helm:

```
helm repo remove {{nome_do_repositório}}
```

- Instala um chart helm:

```
helm install {{nome_do_repositório}}/{{nome_do_chart}}
```

- Obtém um chart helm chart como um arquivo tar:

```
helm get {{nome_do_release_do_chart}}
```

- Atualiza as dependências helm:

```
helm dependency update
```

# hexdump

Imprime dados no formato ASCII, decimal, hexadecimal ou octal.

Mais informações: <https://manned.org/hexdump>.

- Imprimir a representação hexadecimal de um arquivo, substituindo linhas duplicadas por '\*':

```
hexdump {{caminho/para/arquivo}}
```

- Imprimir a representação hexadecimal e ASCII de um arquivo, em duas colunas:

```
hexdump -C {{caminho/para/arquivo}}
```

- Imprimir a representação hexadecimal de um arquivo, porém apresentando apenas seus n primeiros bytes:

```
hexdump -C -n{{numero_de_bytes}} {{caminho/para/arquivo}}
```

- Imprimir a representação hexadecimal completa de um arquivo (sem omitir linhas duplicadas):

```
hexdump --no-squeezing {{caminho/para/arquivo}}
```

# history

Histórico de linha de comando.

Mais informações: [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).

- Exibe a lista de histórico de comandos com números de linha:

```
history
```

- Exibe os últimos 20 comandos (em `zsh` ele exibe todos os comandos a partir do 20º):

```
history {{20}}
```

- Limpa a lista do histórico de comandos (apenas para o shell `bash` atual):

```
history -c
```

- Sobrescreve o arquivo de histórico com o histórico do shell `bash` atual (frequentemente combinado com `history -c` para limpar o histórico):

```
history -w
```

- Deleta a entrada do histórico no deslocamento especificado:

```
history -d {{deslocamento}}
```

# htop

Exibe informação dinâmica em tempo real acerca de processos em execução. Uma versão melhorada do comando **top**.

Mais informações: <https://htop.dev/>.

- Inicializa **htop**:

```
htop
```

- Inicializa **htop** mostrando somente processos pertencentes a um usuário:

```
htop --user {{nome_usuario}}
```

- Ordena processos por um **item\_de\_ordenação** (utilize `htop --sort help` para ver as opções disponíveis):

```
htop --sort {{item_de_ordenacao}}
```

- Vê comandos interativos enquanto roda **htop**:

```
?
```

- Muda para uma aba diferente:

```
tab
```

- Mostra ajuda:

```
htop --help
```



# hx

Este comando é um pseudônimo de **helix**.

- Ver documentação sobre o comando original:

```
tldr helix
```

# id

Exibe o usuário atual e identidade do grupo.

Mais informações: <https://www.gnu.org/software/coreutils/id>.

- Exibe o ID do usuário atual (UID), ID do grupo (GID) e grupos aos quais eles pertencem:

```
id
```

- Exibe a identidade do usuário atual como um número:

```
id -u
```

- Exibe a identidade do grupo atual como um número:

```
id -g
```

- Exibe o ID de um usuário arbitrário (UID), ID do grupo (GID) e grupos aos quais eles pertencem:

```
id {{nomedeusuario}}
```

# img2pdf

Ferramenta de conversão sem perdas de imagens para PDF.

Mais informações: <https://gitlab.mister-muffin.de/josch/img2pdf>.

- Converter múltiplas imagens para um único PDF, cada imagem sendo uma página:

```
img2pdf {{caminho/da/imagem1.jpg}} {{caminho/da/imagem2.jpg}}  
--output {{caminho/do/arquivo.pdf}}
```

- Converter para PDF apenas o primeiro quadro de uma imagem com múltiplos quadros:

```
img2pdf {{caminho/do/arquivo.gif}} --first-frame-only --  
output {{caminho/do/arquivo.pdf}}
```

- Auto-orientar a imagem, usar uma página A4 em modo paisagem e uma borda de 2cm horizontalmente e 5.1cm verticalmente:

```
img2pdf {{caminho/do/arquivo.jpg}} --auto-orient --pagesize  
{{A4^T}} --border {{2cm}}:{{5.1cm}} --output {{caminho/do/  
arquivo.pdf}}
```

- Encolher apenas imagens maiores para um retângulo de 10cm por 15cm dentro de uma página de 30x20cm:

```
img2pdf {{caminho/do/arquivo.tiff}} --pagesize {{30cm}}  
x{{20cm}} --imgsize {{10cm}}x{{15cm}} --fit {{shrink}} --  
output {{caminho/do/arquivo.pdf}}
```

- Converter uma imagem para PDF e especificar os metadados do arquivo resultante:

```
img2pdf {{caminho/do/arquivo.png}} --title {{título}} --  
author {{autor}} --creationdate {{1970-01-31}} --keywords  
{{palavra_chave1 palavra_chave2}} --subject {{assunto}} --  
output {{caminho/do/arquivo.pdf}}
```

# imgcat

Utilitário para exibir imagens diretamente na linha de comando.

Requer um terminal compatível, como o iTerm2.

Mais informações: <https://github.com/danielgatis/imgcat>.

- Exibe uma imagem na linha de comando:

```
imgcat {{nome_do_arquivo}}
```

# jar

Compactador de Bibliotecas e Aplicações Java.

Mais informações: <https://docs.oracle.com/javase/tutorial/deployment/jar/basicsindex.html>.

- Arquia recursivamente todos os arquivos do diretório atual em um arquivo .jar:

```
jar cf {{arquivo.jar}} *
```

- Descompacta o arquivo .jar/.war para o diretório atual:

```
jar -xvf {{arquivo.jar}}
```

- Lista o conteúdo do arquivo .jar/.war:

```
jar tf {{caminho/para/arquivo.jar}}
```

- Lista o conteúdo do arquivo .jar/.war com mais detalhes (verbose):

```
jar tvf {{caminho/para/arquivo.jar}}
```

# java

Inicializador de programas Java.

Mais informações: <https://docs.oracle.com/en/java/javase/20/docs/specs/man/java.html>.

- Executa um arquivo Java `.class` que contém um método `main`, usando o nome da classe:

```
java {{nome_da_classe}}
```

- Executa um programa Java e usa classes adicionais de terceiros ou definidas pelo usuário:

```
java -classpath {{caminho/para/classes1}}:{{caminho/para/classes2}}:. {{nome_da_classe}}
```

- Executa um programa `.jar`:

```
java -jar {{nome_do_arquivo.jar}}
```

- Executa um programa `.jar` com o debugger aguardando conexão na porta 5005:

```
java -  
agentlib:jdwp=transport=dt_socket,server=y,suspend=y,address=*:  
5005 -jar {{nome_do_arquivo.jar}}
```

- Exiba a versão do JDK, JRE e HotSpot:

```
java -version
```

- Exiba os comandos disponíveis do Java:

```
java -help
```

# javac

O compilador de aplicações Java.

Mais informações: <https://docs.oracle.com/en/java/javase/20/docs/specs/man/javac.html>.

- Compile um arquivo .java:

```
javac {{arquivo.java}}
```

- Compile vários arquivos .java:

```
javac {{arquivo1.java}} {{arquivo2.java}} {{arquivo3.java}}
```

- Compile todos os arquivos .java no diretório atual:

```
javac {{*.java}}
```

- Compile um arquivo .java e coloque a classe resultante em um diretório específico:

```
javac -d {{caminho/para/diretorio}} {{arquivo.java}}
```

# join

Junta linhas de dois arquivos ordenados em um campo comum.

Mais informações: <https://www.gnu.org/software/coreutils/join>.

- Junta dois arquivos no primeiro campo (padrão):

```
join {{arquivo1}} {{arquivo2}}
```

- Junta dois arquivos usando uma vírgula (em vez de um espaço) como separador de campo:

```
join -t '{{',''}} {{arquivo1}} {{arquivo2}}
```

- Junta campo3 do arquivo1 ao campo1 do arquivo2:

```
join -1 {{3}} -2 {{1}} {{arquivo1}} {{arquivo2}}
```

- Produz uma linha para cada linha que não pode ser pareada para o arquivo1:

```
join -a {{1}} {{arquivo1}} {{arquivo2}}
```



# jupyter

Aplicação web para criar e compartilhar documentos que contem código, visualizações e anotações.

Usado principalmente para análise de dados, computação científica e aprendizado de máquinas (machine learning).

Mais informações: <https://jupyter.org>.

- Inicia um servidor de notebooks Jupyter no diretório atual:

```
jupyter notebook
```

- Abre um caderno Jupyter específico:

```
jupyter notebook {{exemplo.ipynb}}
```

- Exporta um caderno Jupyter específico para outro formato:

```
jupyter nbconvert --to {{html|markdown|pdf|script}}  
{{exemplo.ipynb}}
```

- Inicia um servidor em uma porta específica:

```
jupyter notebook --port={{porta}}
```

- Lista de servidores de notebooks atualmente em funcionamento:

```
jupyter notebook list
```

- Para o servidor atualmente em funcionamento:

```
jupyter notebook stop
```

- Inicia o JupyterLab, se instalado, no diretório atual:

```
jupyter lab
```

# jwt

Uma ferramenta de linha de comando (command-line tool) para trabalhar com JSON Web Tokens (JWTs).

Algoritmos de encriptação disponíveis são HS256, HS384, HS512, RS256, RS384, RS512, ES256, ES384.

Mais informações: <https://github.com/mike-engel/jwt-cli>.

- Decodifica um JWT:

```
jwt decode {{jwt_string}}
```

- Decodifica um JWT em uma JSON string:

```
jwt decode -j {{jwt_string}}
```

- Codifica uma JSON string em um JWT:

```
jwt encode --alg {{HS256}} --secret {{1234567890}}  
'{{json_string}}'
```

- Codifica dados (payload) de um par de chaves (key pair) em um JWT:

```
jwt encode --alg {{HS256}} --secret {{1234567890}} -P  
chave=valor
```

# kafkacat

Este comando é um pseudônimo de **kcat**.

- Ver documentação sobre o comando original:

`tldr kcat`

# kill

Envia um sinal para um processo, geralmente para finalizar o processo.

Todos os sinais exceto pelo SIGKILL e SIGSTOP podem ser interceptados pelo processo para finalizar de forma limpa.

Mais informações: <https://manned.org/kill>.

- Finaliza um programa usando o sinal default SIGTERM (terminate):

```
kill {{id_do_processo}}
```

- Lista todos os nomes dos sinais disponíveis (para serem usados sem o prefixo SIG):

```
kill -l
```

- Finaliza um processo em background:

```
kill %{{id_do_processo}}
```

- Finaliza um programa usando o sinal SIGHUP. Muitos daemons vão recarregar ao invés de finalizar:

```
kill -{{1|HUP}} {{id_do_processo}}
```

- Finaliza um programa usando o sinal SIGINT (interrupt). Isto é tipicamente iniciado pelo usuário ao pressionar Ctrl + C:

```
kill -{{2|INT}} {{id_do_processo}}
```

- Envia sinal para o sistema operacional para finalizar imediatamente o programa (quem não tem chance de capturar o sinal):

```
kill -{{9|KILL}} {{id_do_processo}}
```

- Envia sinal para o sistema operacional para pausar o programa até um sinal SIGCONT ("continue") seja recebido:

```
kill -{{17|STOP}} {{id_do_processo}}
```

- Envia um sinal SIGUSR1 para todos os processos de um dado GID (group id):

```
kill -{{SIGUSR1}} -{{id_do_grupo}}
```

# kubectl

Linha de comando para executar comando em clusters do Kubernetes.

Alguns subcomandos como **kubectl run** tem sua própria documentação de uso.

Mais informações: <https://kubernetes.io/docs/reference/kubectl/>.

- Lista toda a informação sobre um recurso em detalhes:

```
kubectl get {{pod|service|deployment|ingress|...}} -o wide
```

- Atualiza um pod específico com o label 'unhealthy' e o valor 'true':

```
kubectl label pods {{name}} unhealthy=true
```

- Lista todos os recursos de diferentes tipos:

```
kubectl get all
```

- Exibe os usos de recursos (CPU/Memória/Espaço alocado) dos nós ou pods:

```
kubectl top {{pod|node}}
```

- Exibe os endereços dos serviços do master e do cluster:

```
kubectl cluster-info
```

- Exibe uma explicação de um campo específico:

```
kubectl explain {{pods.spec.containers}}
```

- Exibe os logs de um container em um pod ou de um recurso específico:

```
kubectl logs {{pod_name}}
```

- Executa um comando em um pod existente:

```
kubectl exec {{pod_name}} -- {{ls /}}
```

# llvm-ar

Este comando é um pseudônimo de **ar**.

- Ver documentação sobre o comando original:

```
tldr ar
```

# llvm-g++

Este comando é um pseudônimo de **clang++**.

- Ver documentação sobre o comando original:

```
tldr clang++
```

# llvm-gcc

Este comando é um pseudônimo de **clang**.

- Ver documentação sobre o comando original:

```
tldr clang
```



# llvm-nm

Este comando é um pseudônimo de **nm**.

- Ver documentação sobre o comando original:

```
tldr nm
```

# llvm-objdump

Este comando é um pseudônimo de **objdump**.

- Ver documentação sobre o comando original:

```
tldr objdump
```

# llvm-strings

Este comando é um pseudônimo de **strings**.

- Ver documentação sobre o comando original:

```
tldr strings
```

# ls

Lista o conteúdo de um diretório.

Mais informações: <https://www.gnu.org/software/coreutils/ls>.

- Lista arquivos um por linha:

```
ls -l
```

- Lista todos os arquivos, incluindo arquivos ocultos:

```
ls -a
```

- Lista todos os arquivos, com o final / adicionado aos nomes dos diretórios:

```
ls -F
```

- Lista todos os arquivos em formato longo (permissões, dono, tamanho e data de modificação):

```
ls -la
```

- Lista em formato longo com tamanho exibido usando unidades legíveis para humanos (KiB, MiB, GiB):

```
ls -lh
```

- Lista em formato longo ordenados por tamanhos (decrecente):

```
ls -ls
```

- Lista todos os arquivos em formato longo, ordenados por data de modificação (mais antigo primeiro):

```
ls -ltr
```

- Lista apenas diretórios:

```
ls -d */
```

# lsof

Lista arquivos abertos e os seus processos correspondentes.

Nota: Privilégios de administrador (ou sudo) são necessários para listar arquivos abertos por outros.

Mais informações: <https://manned.org/lsof>.

- Localizar os processos que têm um certo arquivo aberto:

```
lsof {{caminho/para/arquivo}}
```

- Localizar o processo que abriu uma porta de internet local:

```
lsof -i :{{porta}}
```

- Mostrar o ID (PID) do processo que abriu um arquivo especificado:

```
lsof -t {{caminho/para/arquivo}}
```

- Listar arquivos abertos por um certo usuário:

```
lsof -u {{nome_usuario}}
```

- Listar arquivos abertos por um certo comando ou processo:

```
lsof -c {{nome_processo_ou_comando}}
```

- Listar arquivos abertos por um certo processo, dado o seu PID:

```
lsof -p {{PID}}
```

- Listar arquivos abertos em um diretório:

```
lsof +D {{caminho/para/diretório}}
```

- Encontrar o processo que está ouvindo uma porta de TCP local:

```
lsof -iTCP:{{porta}} -sTCP:LISTEN
```

# lzcat

Este comando é um pseudônimo de **xz**.

Mais informações: <https://manned.org/lzcat>.

- Ver documentação sobre o comando original:

```
tldr xz
```

# Izma

Este comando é um pseudônimo de **xz**.

Mais informações: <https://manned.org/lzma>.

- Ver documentação sobre o comando original:

```
tldr xz
```

# man

Formata e exibe páginas de manual.

Mais informações: <https://www.man7.org/linux/man-pages/man1/man.1.html>.

- Exibe a página de manual de um comando:

```
man {{comando}}
```

- Exibe a página de manual de um comando da seção 7:

```
man {{7}} {{comando}}
```

- Lista todas as seções disponíveis para um comando:

```
man -f {{comando}}
```

- Exibe o caminho procurado pelas páginas de manual:

```
man --path
```

- Exibe a localização de uma página de manual em vez da própria página de manual:

```
man -w {{comando}}
```

- Exibe a página de manual usando uma localidade específica:

```
man {{comando}} --locale={{localidade}}
```

- Procura páginas de manual contendo um termo de pesquisa:

```
man -k "{{termo_de_pesquisa}}"
```



# meteor

Plataforma Full-Stack de JavaScript para desenvolver aplicações web.

Mais informações: <https://meteor.com>.

- Executar um projeto meteor a partir do seu diretório base em modo de desenvolvimento:

```
meteor
```

- Criar um novo projeto em um diretório específico:

```
meteor create {{caminho/para/diretório}}
```

- Listar todos os pacotes usados pelo projeto:

```
meteor list
```

- Adicionar um pacote ao projeto:

```
meteor add {{nome_pacote}}
```

- Remover um pacote do projeto:

```
meteor remove {{nome_pacote}}
```

- Criar uma build de produção do projeto, no formato tarball, em um diretório específico:

```
meteor build {{caminho/para/diretório}}
```

# mongo

Cliente shell interativo de MongoDB.

Mais informações: <https://docs.mongodb.com/manual/reference/program/mongo>.

- Conectar a uma base de dados:

```
mongo {{base_de_dados}}
```

- Conectar a uma base de dados em um host e porta específicos:

```
mongo --host {{host}} --port {{porta}} {{base_de_dados}}
```

- Conectar a uma base de dados com um usuário específico, uma senha será pedida ao usuário:

```
mongo --username {{usuário}} {{base_de_dados}} --password
```

- Avaliar JavaScript na base de dados:

```
mongo --eval '{{JSON.stringify(db.foo.findOne())}}'
{{base_de_dados}}
```

# mosquitto

Um broker de MQTT.

Mais informações: <https://mosquitto.org/>.

- Inicia mosquitto:

```
mosquitto
```

- Especifica um arquivo de configuração para usar:

```
mosquitto --config-file {{caminho/para/arquivo.conf}}
```

- Escuta em uma porta específica:

```
mosquitto --port {{8883}}
```

- Cria um processo rodando em background:

```
mosquitto --daemon
```

# mscore

Este comando é um pseudônimo de **musescore**.

Mais informações: <https://musescore.org/handbook/command-line-options>.

- Ver documentação sobre o comando original:

```
tldr musescore
```

# mv

Movimentação de arquivos entre diretórios, ou renomeá-los.

Mais informações: <https://www.gnu.org/software/coreutils/mv>.

- Move um arquivo para um diretório arbitrário:

```
mv {{percorso/del/arquivo}} {{percorso/del/destino}}
```

- Move arquivos para outro diretório, mantendo os nomes dos arquivos:

```
mv {{percorso/del/arquivo_1}} {{percorso/del/arquivo_2}} ... {{percorso/del/destino}}
```

- Não requisitar confirmação para sobrescrição de arquivos:

```
mv -f {{percorso/del/arquivo}} {{percorso/del/destino}}
```

- Requisita confirmação para sobrescrição de arquivos, independentemente das permissões de arquivo:

```
mv -i {{percorso/del/arquivo}} {{percorso/del/destino}}
```

- Não sobrescreva arquivos existentes no diretório de destino:

```
mv -n {{percorso/del/arquivo}} {{percorso/del/destino}}
```

- Move os arquivos em modo Verbose, mostrando os arquivos após sua movimentação:

```
mv -v {{percorso/del/arquivo}} {{percorso/del/destino}}
```

# mvn

Ferramenta para a criação e gerenciamento de projetos Java.

Mais informações: <https://maven.apache.org>.

- Compilar um projeto:

```
mvn compile
```

- Criar um artefato de distribuição utilizando o formato especificado no `pom.xml`, por exemplo o formato `jar`:

```
mvn package
```

- Criar um artefato de distribuição sem executar testes unitários:

```
mvn package -DskipTests
```

- Instalar um artefato gerado em um repositório local:

```
mvn install
```

- Apagar artefatos gerados no diretório `target`:

```
mvn clean
```

- Executar as fases `clean` e `package` em um projeto:

```
mvn clean package
```

- Executar as fases `clean` e `package` em um projeto utilizando um perfil:

```
mvn clean -P{{perfil}} package
```

- Executar uma classe que possua o método `main`:

```
mvn exec:java -Dexec.mainClass="{{nome.do.pacote.classe}}" -  
Dexec.args="{{argument1 argument2 ...}}"
```

# mysql

A ferramenta de linha de comando do MySQL.

Mais informações: <https://www.mysql.com/>.

- Conectar a um banco de dados:

```
mysql {{nome_do_banco_de_dados}}
```

- Conectar a um banco de dados (será solicitada a senha de acesso do usuário):

```
mysql -u {{usuário}} --password {{nome_do_banco_de_dados}}
```

- Conectar a um banco de dados disponível em um endereço específico:

```
mysql -h {{endereço_do_banco_de_dados}}  
{{nome_do_banco_de_dados}}
```

- Conectar a um banco de dados utilizando um socket Unix:

```
mysql --socket {{caminho/para/socket.sock}}
```

- Executar todos os comandos de um arquivo SQL em um banco de dados:

```
mysql -e "source {{nome_do_arquivo.sql}}"  
{{nome_do_banco_de_dados}}
```

# mysqldump

Realizar e restaurar backups no MySQL.

Mais informações: <https://dev.mysql.com/doc/refman/en/mysqldump.html>.

- Criar o backup do banco de dados em arquivo de saída (será solicitada a senha de acesso do usuário):

```
mysqldump -u {{usuário}} --password  
{{nome_do_banco_de_dados}} -r {{arquivo_de_saida.sql}}
```

- Restaurar o conteúdo contido no arquivo de backup em banco de dados específico (será solicitada a senha de acesso do usuário):

```
mysql -u {{usuário}} --password -e "source  
{{arquivo_de_backup.sql}}" {{nome_do_banco_de_dados}}
```



# nano

Editor de texto de linha de comando. Um clone melhorado de **Pico**.

Mais informações: <https://nano-editor.org>.

- Inicia o editor:

```
nano
```

- Inicia o editor sem usar arquivos de configuração:

```
nano --ignorercfiles
```

- Abre arquivos específicos, passando para o próximo arquivos ao fechar o anterior:

```
nano {{caminho/para/arquivo1 caminho/para/arquivo2 ...}}
```

- Abre um arquivo e posiciona o cursor na linha e coluna especificadas:

```
nano +{{linha}},{{coluna}} {{caminho/para/arquivo}}
```

- Abre um arquivo e habilita soft wrapping:

```
nano --softwrap {{caminho/para/arquivo}}
```

- Abre um arquivo e indenta novas linhas de acordo com a indentação da linha anterior:

```
nano --autoindent {{caminho/para/arquivo}}
```

- Abre um arquivo e cria um arquivo de backup (caminho/para/arquivo~) ao salvá-lo:

```
nano --backup {{caminho/para/arquivo}}
```

# nativefier

Utilitário para criação de aplicativos para desktop à partir de web sites.

Mais informações: <https://github.com/jiahaog/nativefier>.

- Gerar aplicativo desktop para site:

```
nativefier {{url}}
```

- Gerar aplicativo desktop com nome customizado:

```
nativefier --name {{nome}} {{url}}
```

- Gerar aplicativo desktop usando um ícone PNG customizado:

```
nativefier --icon {{caminho/para/icone.png}} {{url}}
```

# neofetch

Ferramenta CLI para exibir informações sobre seu sistema operacional, software e hardware.

Mais informações: <https://github.com/dylananaraps/neofetch>.

- Volta à configuração padrão, ou crie-a se for a primeira vez que o programa for executado:

```
neofetch
```

- Habilita/Desabilita uma linha de informação da saída, onde 'info name' é o nome da função no arquivo de configuração, por exemplo, 'memory':

```
neofetch --{{enable|disable}} {{info name}}
```

- Oculta/Mostra arquitetura do SO:

```
neofetch --os_arch {{on|off}}
```

- Habilita/Desabilita a marca da CPU na saída:

```
neofetch --cpu_brand {{on|off}}
```

# next

Framework React que usa renderização do lado do servidor para construir aplicações web otimizadas.

Mais informações: <https://nextjs.org/docs>.

- Inicia a aplicação atual em modo de desenvolvimento:

```
next dev
```

- Inicia a aplicação atual e escuta em uma porta específica:

```
next dev --port {{porta}}
```

- Cria a aplicação atual otimizada para produção:

```
next build
```

- Inicia a aplicação compilada em modo de produção:

```
next start
```

- Inicia a aplicação compilada e escuta em uma porta específica:

```
next start --port {{porta}}
```

- Exporta a aplicação atual para páginas HTML estáticas:

```
next export
```

- Exibe o status de telemetria Next.js:

```
next telemetry
```

- Exibe ajuda para um subcomando:

```
next {{build|dev|export|start|telemetry}} --help
```

# nm-classic

Este comando é um pseudônimo de **nm**.

- Ver documentação sobre o comando original:

```
tldr nm
```

# node

Plataforma de JavaScript para o lado do Servidor (Node.js).

Mais informações: <https://nodejs.org>.

- Executar um arquivo JavaScript:

```
node {{arquivo}}.js
```

- Inicializar a REPL (shell interativa):

```
node
```

- Executar JavaScript, passando-o no comando:

```
node -e "{{código}}"
```

- Executar um arquivo JavaScript, imprimindo o resultado:

```
node -p "{{script}}"
```

- Ativar o inspetor, pausando a execução até que um depurador seja conectado depois que o código-fonte for totalmente analisado:

```
node --no-lazy --inspect-brk {{caminho/para/arquivo}}
```

# nth

Name That Hash - Nomeia o tipo de um hash em um instante.

Mais informações: <https://github.com/hashpals/name-that-hash>.

- Nomeia o hash:

```
nth -t {{5f4dcc3b5aa765d61d8327deb882cf99}}
```

- Nomeia hashes em um arquivo:

```
nth -f {{caminho/para/hashes}}
```

- Saída no formato json:

```
nth -t {{5f4dcc3b5aa765d61d8327deb882cf99}} -g
```

- Decodifica o hash em Base64 e nomeia:

```
nth -t {{NWY0ZGNjM2I1YWE3NjVkJFk0ODMyN2RlYjg4MmNmOTkK}} -b64
```

# ntl

Este comando é um pseudônimo de **netlify**.

Mais informações: <https://cli.netlify.com>.

- Ver documentação sobre o comando original:

```
tldr netlify
```



# nvim

Neovim, um editor de texto para programadores baseado no Vim, oferece vários modos para diferentes tipos de manipulação de texto.

Pressionar **i** no modo normal entra no modo de inserção. **<Esc>** retorna ao modo normal, que não permite a inserção regular de texto.

Veja também **vim**, **vimtutor**, **vimdiff**.

Mais informações: <https://neovim.io>.

- Abre um arquivo:

```
nvim {{caminho/para/arquivo}}
```

- Entra no modo de edição de texto (mode de inserção):

```
<Esc>i
```

- Copia ("yank") ou recorta ("delete") a linha atual (cole-a com P):

```
<Esc>{{yy|dd}}
```

- Entra no modo normal e desfaz a última operação:

```
<Esc>u
```

- Procura por um padrão em um arquivo (pressione n/ N para ir para a próxima/prévia correspondência):

```
<Esc>/{{padrão_procurado}}<Enter>
```

- Executa uma substituição de expressão regular em todo o arquivo:

```
<Esc>:%s/{{expressão_regulard}}/{{substituição}}/g<Enter>
```

- Entra no modo normal, salva (grava) o arquivo e sai:

```
<Esc>:wq<Enter>
```

- Sai sem salvar:

```
<Esc>:q!<Enter>
```

# passwd

Passwd é uma ferramenta usada para alterar a senha de um usuário.

Mais informações: <https://manned.org/passwd>.

- Altera a senha do usuário atual interativamente:

```
passwd
```

- Altera a senha de um usuário específico:

```
passwd {{nome do usuário}}
```

- Obtém o status atual do usuário:

```
passwd -S
```

- Deixa a senha da conta em branco (isso definirá a conta nomeada como sem senha):

```
passwd -d
```

# pdftk

Conjunto de utilitários para manipular arquivos PDF.

Mais informações: <https://www.pdflabs.com/tools/pdftk-the-pdf-toolkit>.

- Extrair conjuntos de páginas de um arquivo PDF (páginas 1 a 3, 5 e 6 a 10) e guardá-las num novo arquivo:

```
pdftk {{arquivo.pdf}} cat {{1-3 5 6-10}} output  
{{novo_arquivo.pdf}}
```

- Concatenar uma lista de arquivos PDF, guardando o resultado num novo arquivo:

```
pdftk {{arquivo1.pdf arquivo2.pdf arquivoN.pdf ...}} cat  
output {{novo_arquivo.pdf}}
```

- Partir cada página de um arquivo PDF num arquivo separado, com um padrão para o nome dos novos arquivos:

```
pdftk {{arquivo.pdf}} burst output {{página_%d.pdf}}
```

- Girar em 180° todas as páginas de um arquivo PDF:

```
pdftk {{arquivo.pdf}} cat {{1-endsouth}} output  
{{novo_arquivo.pdf}}
```

- Girar a terceira página de um arquivo PDF em 90° no sentido horário, não modificando as restantes:

```
pdftk {{arquivo.pdf}} cat {{1-2 3east 4-end}} output  
{{novo_arquivo.pdf}}
```

# pio-init

Este comando é um pseudônimo de **pio project**.

- Ver documentação sobre o comando original:

```
tldr pio project
```

# piodebuggdb

Este comando é um pseudônimo de **pio debug**.

- Ver documentação sobre o comando original:

```
tldr pio debug
```

# pip install

Instala pacotes Python.

Mais informações: <https://pip.pypa.io>.

- Instala um pacote:

```
pip install {{nome_pacote}}
```

- Instala uma versão específica de um pacote:

```
pip install {{nome_pacote}}=={{versão_do_pacote}}
```

- Instala pacotes listados em um arquivo:

```
pip install --requirement {{requirements.txt}}
```

- Instala pacotes a partir de uma URL ou arquivo local (.tar.gz | .whl):

```
pip install --find-links {{url|caminho/do/arquivo}}
```

- Instala o pacote local no diretório atual no modo de desenvolvimento (editável):

```
pip install --editable {{.}}
```

# pip uninstall

Desinstala pacotes Python.

Mais informações: <https://pip.pypa.io>.

- Desinstala um pacote:

```
pip uninstall {{nome_pacote}}
```

- Desinstala pacotes listados em um arquivo:

```
pip uninstall --requirement {{requirements.txt}}
```

- Desinstala um pacote sem pedir por confirmação:

```
pip uninstall --yes {{nome_pacote}}
```

# pip

Gerenciador de pacotes para Python.

Alguns sub-comandos, como **pip install** possuem sua própria documentação.

Mais informações: <https://pip.pypa.io>.

- Instala um pacote (veja `pip install` para mais exemplos de instalação):

```
pip install {{nome_pacote}}
```

- Instala um pacote no diretório do usuário em vez do local padrão do sistema:

```
pip install --user {{nome_pacote}}
```

- Atualiza um pacote:

```
pip install --upgrade {{nome_pacote}}
```

- Desinstala um pacote:

```
pip uninstall {{nome_pacote}}
```

- Salva os pacotes instalados em um arquivo:

```
pip freeze > {{requirements.txt}}
```

- Mostra informações sobre um pacote instalado:

```
pip show {{nome_pacote}}
```

- Instala pacotes a partir de um arquivo:

```
pip install --requirement {{requirements.txt}}
```



# pip3

Gerenciador de pacotes Python.

Mais informações: <https://pip.pypa.io>.

- Instala um pacote:

```
pip3 install {{nome_pacote}}
```

- Instala uma versão específica de um pacote:

```
pip3 install {{nome_pacote}}=={{versão_pacote}}
```

- Atualiza um pacote:

```
pip3 install --upgrade {{nome_pacote}}
```

- Desinstala um pacote:

```
pip3 uninstall {{nome_pacote}}
```

- Salva a lista de pacotes instalados em um arquivo:

```
pip3 freeze > {{requirements.txt}}
```

- Instala pacotes salvos em um arquivo:

```
pip3 install --requirement {{requirements.txt}}
```

- Mostra informações sobre um pacote instalado:

```
pip3 show {{nome_pacote}}
```

# platformio

Este comando é um pseudônimo de **pio**.

Mais informações: <https://docs.platformio.org/en/latest/core/userguide/>.

- Ver documentação sobre o comando original:

```
tldr pio
```

# podman build

Ferramenta sem daemon para criar imagens de contêiner.

O Podman fornece uma linha de comando comparável ao Docker-CLI.

Simplificando: **alias docker=podman**.

Mais informações: <https://docs.podman.io/en/latest/markdown/podman-build.1.html>.

- Criar uma imagem usando um `Dockerfile` ou `Containerfile` no diretório especificado:

```
podman build {{caminho/para/diretório}}
```

- Criar uma imagem com uma tag especificada:

```
podman build --tag {{nome_da_imagem:versão}} {{caminho/para/diretório}}
```

- Criar uma imagem a partir de um arquivo não padrão:

```
podman build --file {{Containerfile.diferente}} .
```

- Criar uma imagem sem usar nenhuma imagem em cache previamente:

```
podman build --no-cache {{caminho/para/diretório}}
```

- Criar uma imagem suprimindo todas as saídas:

```
podman build --quiet {{caminho/para/diretório}}
```

# podman-compose

Executar e gerenciar definição de contêineres Compose Specification.

Mais informações: <https://github.com/containers/podman-compose>.

- Listar todos os contêineres em execução:

```
podman-compose ps
```

- Criar e iniciar todos os contêineres em segundo plano usando um arquivo `docker-compose.yml` local:

```
podman-compose up -d
```

- Iniciar todos os contêineres, fazendo o build se necessário:

```
podman-compose up --build
```

- Iniciar todos os contêineres usando um arquivo de composição alternativo:

```
podman-compose {{caminho/para/arquivo}} up
```

- Parar todos os contêineres em execução:

```
podman-compose stop
```

- Remover todos os contêineres, redes e volumes:

```
podman-compose down --volumes
```

- Acompanhar logs de um contêiner (omitir todos os nomes de contêineres):

```
podman-compose logs --follow {{nome_do_contêiner}}
```

- Executar um comando único em um serviço sem mapear portas:

```
podman-compose run {{nome_do_serviço}} {{comando}}
```

# podman image

Gerenciar imagens Docker.

Veja também **podman build**, **podman import** e **podman pull**.

Mais informações: <https://docs.podman.io/en/latest/markdown/podman-image.1.html>.

- Listar imagens Docker locais:

```
podman image ls
```

- Excluir imagens Docker locais não utilizadas:

```
podman image prune
```

- Excluir todas as imagens não utilizadas (não apenas aquelas sem uma tag):

```
podman image prune --all
```

- Mostrar o histórico de uma imagem Docker local:

```
podman image history {{imagem}}
```

# podman images

Gerenciar imagens do Podman.

Mais informações: <https://docs.podman.io/en/latest/markdown/podman-images.1.html>.

- Listar todas as imagens do Podman:

```
podman images
```

- Listar todas as imagens do Podman, incluindo intermediárias:

```
podman images --all
```

- Listar a saída no modo silencioso (apenas IDs numéricos):

```
podman images --quiet
```

- Listar todas as imagens do Podman que não são utilizadas por nenhum contêiner:

```
podman images --filter dangling=true
```

- Listar imagens que contenham uma substring em seus nomes:

```
podman images "{{*imagem|imagem*}}"
```

# podman machine

Criar e gerenciar máquinas virtuais executando o Podman.

Incluído com a versão 4 ou superior do Podman.

Mais informações: <https://docs.podman.io/en/latest/markdown/podman-machine.1.html>.

- Listar as máquinas existentes:

```
podman machine ls
```

- Criar uma nova máquina padrão:

```
podman machine init
```

- Criar uma nova máquina com um nome específico:

```
podman machine init {{nome}}
```

- Criar uma nova máquina com recursos diferentes:

```
podman machine init --cpus={{4}} --memory={{4096}} --disk-size={{50}}
```

- Iniciar ou parar uma máquina:

```
podman machine {{start|stop}} {{nome}}
```

- Conectar-se a uma máquina em execução via SSH:

```
podman machine ssh {{nome}}
```

- Inspecionar informações sobre uma máquina:

```
podman machine inspect {{nome}}
```

# podman ps

Listar contêineres do Podman.

Mais informações: <https://docs.podman.io/en/latest/markdown/podman-ps.1.html>.

- Listar contêineres do Podman em execução atualmente:

```
podman ps
```

- Listar todos os contêineres do Podman (em execução e parados):

```
podman ps --all
```

- Mostrar o contêiner mais recente criado (inclui todos os estados):

```
podman ps --latest
```

- Filtrar contêineres que contenham uma substring em seus nomes:

```
podman ps --filter "name={{nome}}"
```

- Filtrar contêineres que compartilhem uma determinada imagem como ancestral:

```
podman ps --filter "ancestor={{imagem}}:{{tag}}"
```

- Filtrar contêineres pelo código de status de saída:

```
podman ps --all --filter "exited={{código}}"
```

- Filtrar contêineres pelo status (criado, em execução, removendo, pausado, encerrado e morto):

```
podman ps --filter "status={{status}}"
```

- Filtrar contêineres que montam um volume específico ou têm um volume montado em um caminho específico:

```
podman ps --filter "volume={{caminho/para/diretório}}" --format "table {{.ID}}\t{{.Image}}\t{{.Names}}\t{{.Mounts}}"
```



# podman rmi

Remover uma ou mais imagens do Podman.

Mais informações: <https://docs.podman.io/en/latest/markdown/podman-rmi.1.html>.

- Remover uma ou mais imagens pelo nome delas:

```
podman rmi {{imagem:tag}} {{imagem2:tag}} {{...}}
```

- Remover uma imagem forçadamente:

```
podman rmi --force {{imagem}}
```

- Remover uma imagem sem excluir os pais não marcados:

```
podman rmi --no-prune {{imagem}}
```

- Exibir ajuda:

```
podman rmi
```

# podman run

Executar um comando em um novo contêiner Podman.

Mais informações: <https://docs.podman.io/en/latest/markdown/podman-run.1.html>.

- Executar um comando em um novo contêiner a partir de uma imagem marcada:

```
podman run {{imagem:tag}} {{comando}}
```

- Executar um comando em um novo contêiner em segundo plano e exibir o ID:

```
podman run --detach {{imagem:tag}} {{comando}}
```

- Executar um comando em um contêiner temporário no modo interativo e pseudo-TTY:

```
podman run --rm --interactive --tty {{imagem:tag}}  
{{comando}}
```

- Executar um comando em um novo contêiner com variáveis de ambiente passadas:

```
podman run --env '{{variável}}={{valor}}' --env {{variável}}  
{{imagem:tag}} {{comando}}
```

- Executar um comando em um novo contêiner com volumes montados por bind:

```
podman run --volume {{/caminho/para/caminho_no_host}}:{{/  
caminho/para/caminho_no_contêiner}} {{imagem:tag}}  
{{comando}}
```

- Executar um comando em um novo contêiner com portas publicadas:

```
podman run --publish {{porta_no_host}}:{{porta_no_contêiner}}  
{{imagem:tag}} {{comando}}
```

- Executar um comando em um novo contêiner sobrescrevendo o ponto de entrada (entrypoint) da imagem:

```
podman run --entrypoint {{comando}} {{imagem:tag}}
```

- Executar um comando em um novo contêiner conectando-o a uma rede:

```
podman run --network {{rede}} {{imagem:tag}}
```

# podman

Ferramenta de gerenciamento simples para pods, contêineres e imagens.

O Podman fornece uma linha de comando comparável ao Docker-CLI.

Simplificando: **alias docker=podman**.

Mais informações: <https://github.com/containers/podman/blob/main/commands-demo.md>.

- Listar todos os contêineres (em execução e parados):

```
podman ps --all
```

- Criar um contêiner a partir de uma imagem, com um nome personalizado:

```
podman run --name {{nome_do_contêiner}} {{imagem}}
```

- Iniciar ou parar um contêiner existente:

```
podman {{start|stop}} {{nome_do_contêiner}}
```

- Baixar uma imagem de um registro (por padrão, Docker Hub):

```
podman pull {{imagem}}
```

- Exibir a lista de imagens já baixadas:

```
podman images
```

- Abrir um shell dentro de um contêiner que já está em execução:

```
podman exec --interactive --tty {{nome_do_contêiner}} {{sh}}
```

- Remover um contêiner parado:

```
podman rm {{nome_do_contêiner}}
```

- Exibir os logs de um ou mais contêineres e acompanhar a saída do log:

```
podman logs --follow {{nome_do_contêiner}}  
{{id_do_contêiner}}
```

# pssh

Programa de SSH paralelo.

Mais informações: <https://manned.org/pssh>.

- Executar um comando em dois hosts e imprimir a saída em cada servidor em linha:

```
pssh -i -H "{{host1}} {{host2}}" {{hostname -i}}
```

- Executar um comando e salvar a saída em arquivos separados:

```
pssh -H {{host1}} -H {{host2}} -o {{caminho/para/diretório_de_saída}} {{hostname -i}}
```

- Executar um comando em vários hosts, especificados em um arquivo separado por nova linha:

```
pssh -i -h {{caminho/para/arquivo_de_hosts}} {{hostname -i}}
```

- Executar um comando como root (isso solicitará a senha do root):

```
pssh -i -h {{caminho/para/arquivo_de_hosts}} -A -l {{nome_de_usuario_do_root}} {{hostname -i}}
```

- Executar um comando com argumentos SSH adicionais:

```
pssh -i -h {{caminho/para/arquivo_de_hosts}} -x "{{-0 VisualHostKey=yes}}" {{hostname -i}}
```

- Executar um comando limitando o número de conexões paralelas para 10:

```
pssh -i -h {{caminho/para/arquivo_de_hosts}} -p {{10}} '{{cd dir; ./script.sh; exit}}'
```

# ptpython3

Este comando é um pseudônimo de **ptpython**.

- Ver documentação sobre o comando original:

```
tldr ptpython
```

# python

Interpretador da linguagem Python.

Mais informações: <https://www.python.org>.

- Inicia o REPL (shell interativo):

```
python
```

- Executa um arquivo Python específico:

```
python {{caminho/para/arquivo.py}}
```

- Executa um arquivo Python específico e inicia um REPL:

```
python -i {{caminho/para/arquivo.py}}
```

- Executa uma expressão em Python:

```
python -c "{{expressão}}"
```

- Roda o script do módulo de biblioteca especificado:

```
python -m {{modulo}} {{argumentos}}
```

- Instala um pacote usando pip:

```
python -m pip install {{nome_do_pacote}}
```

- Depura interativamente um script de Python:

```
python -m {{pdb}} {{caminho/para/arquivo.py}}
```

- Inicia o servidor HTTP integrado na porta 8000 no diretório atual:

```
python -m {{http.server}}
```

# python3

Este comando é um pseudônimo de **python**.

- Ver documentação sobre o comando original:

`tldr python`

# r2

Este comando é um pseudônimo de **radare2**.

- Ver documentação sobre o comando original:

```
tldr radare2
```



# rails generate

Gerar artefatos Rails a partir de um modelo em um projeto existente.

Mais informações: [https://guides.rubyonrails.org/command\\_line.html#bin-rails-generate](https://guides.rubyonrails.org/command_line.html#bin-rails-generate).

- Exibir todos os geradores disponíveis:

```
rails generate
```

- Criar um modelo:

```
rails generate model {{nome_da_classe}}
```

- Criar um controlador:

```
rails generate controller {{nome_do_controlador}}
```

- Criar uma estrutura completa (modelo, controlador, testes, etc.) para um novo modelo:

```
rails generate scaffold {{nome_do_modelo}}
```

# rails

Framework MVC para o desenvolvimento de aplicações web, desenvolvido em Ruby.

Mais informações: [https://guides.rubyonrails.org/command\\_line.html](https://guides.rubyonrails.org/command_line.html).

- Criar um novo projeto:

```
rails new "{{nome_do_projeto}}"
```

- Iniciar o servidor local para o projeto atual na porta 3000:

```
rails server
```

- Iniciar o servidor local para o projeto atual em uma porta específica:

```
rails server -p "{{porta}}"
```

- Iniciar o console Rails para manipular o projeto atual utilizando o terminal:

```
rails console
```

- Verificar a versão atual do Rails:

```
rails --version
```

# rbenv

Ferramenta que facilita a instalação e gerenciamento de múltiplas versões da linguagem Ruby.

Mais informações: <https://github.com/rbenv/rbenv>.

- Instalar uma ou mais versões, separadas por espaço:

```
rbenv install {{uma_ou_mais_versoes}}
```

- Exibir a lista de versões instaladas:

```
rbenv versions
```

- Determinar uma versão específica para ser a instalação padrão:

```
rbenv global {{versao}}
```

- Determinar uma versão específica para um projeto. Este comando deve ser executado no diretório do projeto:

```
rbenv local {{versao}}
```

- Exibir a versão ativa:

```
rbenv version
```

- Remover uma versão:

```
rbenv uninstall {{versao}}
```

- Exibir todas as versões que contém um determinado executável:

```
rbenv whence {{executavel}}
```

# rcat

Este comando é um pseudônimo de **rc**.

- Ver documentação sobre o comando original:

```
tldr rc
```

# renice

Altera a prioridade/agradabilidade de agendamento de um ou mais processos em execução.

Os valores de agradabilidade variam de -20 (mais favorável ao processo) a 19 (menos favorável ao processo).

Mais informações: <https://manned.org/renice>.

- Altera a prioridade de um processo em execução:

```
renice -n {{valor_de_agradabilidade}} -p {{pid}}
```

- Altera a prioridade de todos os processos pertencentes a um usuário:

```
renice -n {{valor_de_agradabilidade}} -u {{nome_do_usuario}}
```

- Altera a prioridade de todos os processos que pertencem a um grupo de processos:

```
renice -n {{valor_de_agradabilidade}} --pgrp  
{{grupo_de_processos}}
```

# ripgrep

Este comando é um pseudônimo de **rg**.

- Ver documentação sobre o comando original:

```
tldr rg
```

# rsync

Transfira arquivos para ou de um host remote (mas não entre dois hosts remotos), usando SSH por padrão.

Para especificar um caminho remoto, use **host:caminho/para/arquivo\_ou\_diretório**.

Mais informações: <https://download.samba.org/pub/rsync/rsync.1>.

- Transfere um arquivo:

```
rsync {{caminho/para/origem}} {{caminho/para/destino}}
```

- Usa o modo de arquivo (copia recursivamente diretórios, copia links simbólicos sem resolver e preserva permissões, propriedade e tempos de modificação):

```
rsync --archive {{caminho/para/origem}} {{caminho/para/destino}}
```

- Comprime os dados à medida que são enviados ao destino, exibe progresso detalhado e legível, e mantém arquivos parcialmente transferidos se forem interrompidos:

```
rsync --compress --verbose --human-readable --partial --progress {{caminho/para/origem}} {{caminho/para/destino}}
```

- Copia recursivamente diretórios:

```
rsync --recursive {{caminho/para/origem}} {{caminho/para/destino}}
```

- Transfere os conteúdos do diretório, mas não o diretório em si:

```
rsync --recursive {{caminho/para/origem}}/ {{caminho/para/destino}}
```

- Copia recursivamente diretórios, usa o modo de arquivamento, resolve links simbólicos e ignora arquivos que são mais recentes no destino:

```
rsync --recursive --archive --update --copy-links {{caminho/para/origem}} {{caminho/para/destino}}
```

- Transfere um diretório para um host remoto executando o `rsyncd` and exclui arquivos no destino que não existem na origem:

```
rsync --recursive --delete rsync://{{host}}:{{caminho/para/origem}} {{caminho/para/destino}}
```

- Transfere um arquivo por SSH usando uma porta diferente da padrão (22) e mostra o progresso global:

```
rsync --rsh 'ssh -p {{porta}}' --info=progress2 {{host}}:{{caminho/para/origem}} {{caminho/para/destino}}
```



# rustc

O compilador Rust.

Processa, compila e vincula arquivos fonte da linguagem Rust.

Mais informações: <https://doc.rust-lang.org/rustc>.

- Compila um único arquivo:

```
rustc {{caminho/para/arquivo.rs}}
```

- Compila com alta otimização:

```
rustc -O {{caminho/para/arquivo.rs}}
```

- Compila com informações de depuração:

```
rustc -g {{caminho/para/arquivo.rs}}
```

- Compila com otimizações específicas de arquitetura para a CPU atual:

```
rustc -C target-cpu=native {{caminho/para/arquivo.rs}}
```

- Exibe otimizações específicas de arquitetura para a CPU atual:

```
rustc -C target-cpu=native --print cfg
```

- Exibe lista de targets:

```
rustc --print target-list
```

- Compila para um target específico:

```
rustc --target {{target_triplo}} {{caminho/para/arquivo.rs}}
```

# rvm

Ferramenta que facilita a instalação e gerenciamento de múltiplas versões da linguagem Ruby.

Mais informações: <https://rvm.io>.

- Instalar uma ou mais versões separadas por espaço:

```
rvm install {{uma_ou_mais_versoes}}
```

- Exibir a lista de versões instaladas:

```
rvm list
```

- Definir uma versão específica para ser utilizada temporariamente:

```
rvm use {{versao}}
```

- Definir uma versão específica para ser a instalação padrão:

```
rvm --default use {{versao}}
```

- Atualizar uma versão já instalada para uma nova versão:

```
rvm upgrade {{versao_atual}} {{nova_versao}}
```

- Remover uma versão mantendo o código fonte:

```
rvm uninstall {{versao}}
```

- Remover uma versão e o código fonte:

```
rvm remove {{versao}}
```

- Exibir as dependências específicas para o seu sistema operacional:

```
rvm requirements
```

# SCC

Utilitário escrito em GO que conta linhas de código.

Mais informações: <https://github.com/boyter/scc>.

- Mostrar o número de linhas de código no diretório atual:

```
scc
```

- Mostrar o número de linhas de código de um diretório especificado:

```
scc {{caminho/para/diretorio}}
```

- Mostrar o número de linhas de código por arquivo:

```
scc --by-file
```

- Mostrar o resultado usando um formato específico (formato padrão é o tabular):

```
scc --format {{tabular|wide|json|csv|cloc-yaml|html|html-table}}
```

- Contar apenas os arquivos com as extensões especificadas:

```
scc --include-ext {{go, java, js}}
```

- Excluir diretórios da contagem:

```
scc --exclude-dir {{.git,.hg}}
```

- Mostrar output organizado de acordo com o parâmetro especificado (organização padrão é files):

```
scc --sort {{files|name|lines|blanks|code|comments|complexity}}
```

- Mostra a tela de ajuda:

```
scc -h
```

# sh

Bourne shell, o interpretador padrão da linha de comando.

Veja também **histexpand** para um histórico expandido.

Mais informações: <https://manned.org/sh>.

- Inicie uma sessão shell interativa:

```
sh
```

- Execute um comando e então saia da shell:

```
sh -c "{{comando}}"
```

- Execute um script:

```
sh {{caminho/para/script.sh}}
```

- Leia e execute comandos da entrada `stdin` (entrada padrão):

```
sh -s
```

# sl

Locomotiva a vapor passando pelo seu terminal.

Mais informações: <https://github.com/mtoyoda/sl>.

- Faz uma locomotiva a vapor passar pelo seu terminal:

```
sl
```

- O trem pega fogo e pessoas gritam:

```
sl -a
```

- Faz o trem voar:

```
sl -F
```

- Faz o trem pequeno:

```
sl -l
```

- Deixa o usuário sair (CTRL + C):

```
sl -e
```

# snakefmt

Formata arquivos Snakemake.

Mais informações: <https://github.com/snakemake/snakefmt>.

- Formata um Snakefile específico:

```
snakefmt {{caminho/para/snakefile}}
```

- Formata todos os Snakefiles recursivamente em uma pasta específica:

```
snakefmt {{caminho/para/diretorio}}
```

- Formata um arquivo usando um arquivo de configuração específico:

```
snakefmt --config {{caminho/para/config.toml}} {{caminho/para/snakefile}}
```

- Formata um arquivo usando um comprimento máximo de linha específico:

```
snakefmt --line-length {{100}} {{caminho/para/snakefile}}
```

- Exibe às mudanças que seriam realizadas sem realiza-las:

```
snakefmt --diff {{caminho/para/snakefile}}
```

# ssh-add

Gerencia as chaves SSH carregadas no ssh-agent.

Certifique-se de que o ssh-agent esteja em execução para que as chaves sejam carregadas nele.

Mais informações: <https://man.openbsd.org/ssh-add>.

- Adicionar as chaves SSH padrão em ~/.ssh ao ssh-agent:

```
ssh-add
```

- Adicionar uma chave específica ao ssh-agent:

```
ssh-add {{caminho/para/chave_privada}}
```

- Listar as impressões digitais das chaves carregadas atualmente:

```
ssh-add -l
```

- Excluir uma chave do ssh-agent:

```
ssh-add -d {{caminho/para/chave_privada}}
```

- Excluir todas as chaves carregadas atualmente do ssh-agent:

```
ssh-add -D
```

- Adicionar uma chave ao ssh-agent e ao keychain:

```
ssh-add -K {{caminho/para/chave_privada}}
```

# ssh-agent

Iniciar um processo do Agente SSH.

Um Agente SSH mantém as chaves SSH descriptografadas na memória até serem removidas ou o processo ser encerrado.

Veja também **ssh-add**, que pode adicionar e gerenciar chaves mantidas por um Agente SSH.

Mais informações: <https://man.openbsd.org/ssh-agent>.

- Iniciar um Agente SSH para o shell atual:

```
eval $(ssh-agent)
```

- Encerrar o Agente em execução atualmente:

```
ssh-agent -k
```



# ssh-copy-id

Instala a sua chave pública no arquivo `authorized_keys` de uma máquina remota.

Mais informações: <https://manned.org/ssh-copy-id>.

- Copiar suas chaves para a máquina remota:

```
ssh-copy-id {{nome_de_usuario@host_remoto}}
```

- Copiar a chave pública fornecida para o remoto:

```
ssh-copy-id -i {{caminho/para/certificado}}  
{{nome_de_usuario}}@{{host_remoto}}
```

- Copiar a chave pública fornecida para o remoto usando uma porta específica:

```
ssh-copy-id -i {{caminho/para/certificado}} -p {{porta}}  
{{nome_de_usuario}}@{{host_remoto}}
```

# ssh-keygen

Gera chaves SSH usadas para autenticação, logins sem senha e outras finalidades.

Mais informações: <https://man.openbsd.org/ssh-keygen>.

- Gerar uma chave interativamente:

```
ssh-keygen
```

- Gerar uma chave ed25519 com 32 rounds de função de derivação de chave e salvar a chave em um arquivo específico:

```
ssh-keygen -t {{ed25519}} -a {{32}} -f {{~/ .ssh/  
nome_do_arquivo}}
```

- Gerar uma chave RSA de 4096 bits com um comentário de email:

```
ssh-keygen -t {{rsa}} -b {{4096}} -C "{{comentário|email}}"
```

- Remover as chaves de um host do arquivo known\_hosts (útil quando um host conhecido tem uma nova chave):

```
ssh-keygen -R {{host_remoto}}
```

- Obter a impressão digital de uma chave em MD5 Hex:

```
ssh-keygen -l -E {{md5}} -f {{~/ .ssh/nome_do_arquivo}}
```

- Alterar a senha de uma chave:

```
ssh-keygen -p -f {{~/ .ssh/nome_do_arquivo}}
```

- Alterar o tipo de formato da chave (por exemplo, de formato OPENSSH para PEM), o arquivo será reescrito no local:

```
ssh-keygen -p -N "" -m {{PEM}} -f {{~/ .ssh/  
chave_privada_OpenSSH}}
```

- Obter a chave pública a partir da chave secreta:

```
ssh-keygen -y -f {{~/ .ssh/chave_privada_OpenSSH}}
```

# ssh-keyscan

Obter as chaves públicas SSH de hosts remotos.

Mais informações: <https://man.openbsd.org/ssh-keyscan>.

- Obter todas as chaves públicas SSH de um host remoto:

```
ssh-keyscan {{host}}
```

- Obter todas as chaves públicas SSH de um host remoto que esteja ouvindo em uma porta específica:

```
ssh-keyscan -p {{porta}} {{host}}
```

- Obter determinados tipos de chaves públicas SSH de um host remoto:

```
ssh-keyscan -t {{rsa,dsa,ecdsa,ed25519}} {{host}}
```

- Atualizar manualmente o arquivo known\_hosts do SSH com a impressão digital de um determinado host:

```
ssh-keyscan -H {{host}} >> ~/.ssh/known_hosts
```

# ssh

O Secure Shell é um protocolo usado para fazer login de forma segura em sistemas remotos.

Ele pode ser usado para fazer login ou executar comandos em um servidor remoto.

Mais informações: <https://man.openbsd.org/ssh>.

- Conectar-se a um servidor remoto:

```
ssh {{nome_do_usuario}}@{{host_remoto}}
```

- Conectar-se a um servidor remoto com uma identidade específica (chave privada):

```
ssh -i {{caminho/para/arquivo_de_chave}} {{nome_do_usuario}}@{{host_remoto}}
```

- Conectar-se a um servidor remoto usando uma porta específica:

```
ssh {{nome_do_usuario}}@{{host_remoto}} -p {{2222}}
```

- Executar um comando em um servidor remoto com uma alocação de [t]ty permitindo interação com o comando remoto:

```
ssh {{nome_do_usuario}}@{{host_remoto}} -t {{comando}}  
{{argumentos_do_comando}}
```

- Tunelamento SSH: Encaminhamento dinâmico de porta (proxy SOCKS em localhost:1080):

```
ssh -D {{1080}} {{nome_do_usuario}}@{{host_remoto}}
```

- Tunelamento SSH: Encaminhar uma porta específica (localhost:9999 para example.org:80), desativar alocação de pseudo-[t]ty e execução de comandos remotos:

```
ssh -L {{9999}}:{{example.org}}:{{80}} -N -T  
{{nome_do_usuario}}@{{host_remoto}}
```

- Saltar com SSH: Conectar-se a um servidor remoto através de um host intermediário (vários saltos intermediários podem ser especificados separados por vírgula):

```
ssh -J {{nome_do_usuario}}@{{host_intermediário}}  
{{nome_do_usuario}}@{{host_remoto}}
```

- Encaminhamento do agente: Encaminhar as informações de autenticação para a máquina remota (consulte `man ssh_config` para opções disponíveis):

```
ssh -A {{nome_do_usuario}}@{{host_remoto}}
```

# sshd

Daemon do Secure Shell - permite que máquinas remotas façam login de forma segura na máquina atual.

Máquinas remotas podem executar comandos como se estivessem sendo executados nesta máquina.

Mais informações: <https://man.openbsd.org/sshd>.

- Iniciar o daemon em segundo plano:

```
sshd
```

- Executar o sshd em primeiro plano:

```
sshd -D
```

- Executar com saída detalhada (para depuração):

```
sshd -D -d
```

- Executar em uma porta específica:

```
sshd -p {{porta}}
```

# sshfs

Cliente de sistema de arquivos baseado em SSH.

Mais informações: <https://github.com/libfuse/sshfs>.

- Montar um diretório remoto:

```
sshfs {{nome_do_usuario}}@{{host_remoto}}:
{{diretorio_remoto}} {{ponto_de_montagem}}
```

- Desmontar um diretório remoto:

```
umount {{ponto_de_montagem}}
```

- Montar um diretório remoto de um servidor com uma porta específica:

```
sshfs {{nome_do_usuario}}@{{host_remoto}}:
{{diretorio_remoto}} -p {{2222}}
```

- Usar compressão:

```
sshfs {{nome_do_usuario}}@{{host_remoto}}:
{{diretorio_remoto}} -C
```

- Seguir links simbólicos:

```
sshfs -o follow_symlinks {{nome_do_usuario}}@{{host_remoto}}:
{{diretorio_remoto}} {{ponto_de_montagem}}
```

# sshpass

Um provedor de senha SSH.

Ele funciona criando um TTY, inserindo a senha nele e, em seguida, redirecionando **stdin** para a sessão SSH.

Mais informações: <https://manned.org/sshpass>.

- Conectar-se a um servidor remoto usando uma senha fornecida em um descritor de arquivo (neste caso, `stdin`):

```
sshpass -d {{0}} ssh {{usuário}}@{{hostname}}
```

- Conectar-se a um servidor remoto com a senha fornecida como opção e aceitar automaticamente chaves SSH desconhecidas:

```
sshpass -p {{senha}} ssh -o StrictHostKeyChecking=no  
{{usuário}}@{{hostname}}
```

- Conectar-se a um servidor remoto usando a primeira linha de um arquivo como senha, aceitar automaticamente chaves SSH desconhecidas e executar um comando:

```
sshpass -f {{caminho/para/arquivo}} ssh -o  
StrictHostKeyChecking=no {{usuário}}@{{hostname}}  
"{{comando}}"
```



# sudo

Executa um único comando como o Superuser, ou como outro usuário.

Mais informações: <https://www.sudo.ws/sudo.html>.

- Executa um comando como Superuser:

```
sudo {{less /var/log/syslog}}
```

- Edita um arquivo, como Superuser, com seu editor padrão:

```
sudo --edit {{/etc/fstab}}
```

- Executa um comando como outro usuário e/ou grupo:

```
sudo --user={{usuário}} --group={{grupo}} {{id -a}}
```

- Executa um comando anterior com o prefixo `sudo` (apenas em `bash`, `zsh`, etc.):

```
sudo !!
```

- Abre o shell padrão com privilégios de Superuser e executar arquivos de login (`.profile`, `.bash_profile`, etc.):

```
sudo --login
```

- Abre o shell padrão com privilégios de Superuser sem alterar o ambiente de execução:

```
sudo --shell
```

- Abre o shell padrão como dado usuário, carregando o ambiente de execução deste usuário e lendo arquivos de login (`.profile`, `.bash_profile`, etc.):

```
sudo --login --user={{usuário}}
```

- Lista os comandos permitidos (e não permitidos) para o usuário atual:

```
sudo --list
```

# tar

Ferramenta de compressão de arquivos.

Utilizado com metodos de compressão como o de gzip ou bzip2.

Mais informações: <https://www.gnu.org/software/tar>.

- Compactando arquivos em um arquivo tar:

```
tar -cvf {{output.tar}} {{arquivo1}} {{arquivo2}}  
{{arquivo3}}
```

- Compactando arquivos em arquivo gzip:

```
tar -czvf {{output.tar.gz}} {{arquivo1}} {{arquivo2}}  
{{arquivo3}}
```

- Compactando arquivos definindo tipo de compressão automaticamente por extensão:

```
tar -cavf {{output.tar.xz}} {{arquivo1}} {{arquivo2}}  
{{arquivo3}}
```

- Extraíndo arquivos de um arquivo compactado:

```
tar -xvf {{input.tar[.gz|.bz2|.xz]}}
```

- Extraíndo arquivos de um arquivo compactado filtrando por gzip:

```
tar -xzvf {{input.tar[.gz|.bz2|.xz]}}
```

- Extraíndo arquivos de um arquivo compactado para um diretório específico:

```
tar -xvf {{input.tar[.gz|.bz2|.xz]}} -C {{diretório}}
```

- Extrair arquivos seguindo um padrão:

```
tar -xvf {{input.tar}} --wildcards "{{*.html}}"
```

- Listando arquivos de um arquivo tar:

```
tar -tvf {{input.tar}}
```

# tldr

Exibe páginas de ajuda simples para ferramentas de linha de comando do projeto tldr-pages.

Mais informações: <https://tldr.sh>.

- Imprime a página do tldr para um comando específico (dica: é assim que você chegou aqui!):

```
tldr {{comando}}
```

- Imprime a página do tldr para um subcomando específico:

```
tldr {{comando}}-{{subcomando}}
```

- Imprime a página do tldr para um comando para uma [p]lataforma específica:

```
tldr -p {{android|linux|osx|sunos|windows}} {{comando}}
```

- Atualiza o cache local das páginas do tldr:

```
tldr -u
```

# tldr

Este comando é um pseudônimo de **tldr-lint**.

Mais informações: <https://github.com/tldr-pages/tldr-lint>.

- Ver documentação sobre o comando original:

```
tldr tldr-lint
```

# tlmgr-arch

Este comando é um pseudônimo de **tlmgr platform**.

Mais informações: <https://www.tug.org/texlive/tlmgr.html>.

- Ver documentação sobre o comando original:

```
tldr tlmgr platform
```

# todoman

Este comando é um pseudônimo de **todo**.

Mais informações: <https://todoman.readthedocs.io/>.

- Ver documentação sobre o comando original:

```
tldr todo
```

# touch

Cria arquivos e define tempo de acesso/modificação.

Mais informações: <https://manned.org/man/freebsd-13.1/touch>.

- Cria arquivos especificados:

```
touch {{caminho/para/arquivo1 caminho/para/arquivo2 ...}}
```

- Define o tempo de [a]cesso ou [m]odificação do arquivo como o atual e não [c]ria o arquivo se ele não existir:

```
touch -c -{{a|m}} {{caminho/para/arquivo1 caminho/para/arquivo2 ...}}
```

- Define o [t]empo do arquivo para um valor especificado e não [c]ria o arquivo se ele não existir:

```
touch -c -t {{YYYYMMDDHHMM.SS}} {{caminho/para/arquivo1  
caminho/para/arquivo2 ...}}
```

- Define o tempo de um arquivo específico para o tempo de out[r]o arquivo e não [c]ria o arquivo se ele não existir:

```
touch -c -r {{~/ .emacs}} {{caminho/para/arquivo1 caminho/  
para/arquivo2 ...}}
```

# transmission

Este comando é um pseudônimo de **transmission-daemon**.

Mais informações: <https://transmissionbt.com/>.

- Ver documentação sobre o comando original:

```
tldr transmission-daemon
```



# tree

Exibe o conteúdo do diretório atual em formato de árvore.

Mais informações: <http://mama.indstate.edu/users/ice/tree/>.

- Exibe os arquivos e diretórios de acordo com o nível de profundidade 'num' informado (onde 1 significa o diretório atual):

```
tree -L {{num}}
```

- Exibe apenas diretórios:

```
tree -d
```

- Inclui a exibição de arquivos ocultos com colorização diferenciada:

```
tree -a -C
```

- Exibe a árvore sem indentação, mostrando o caminho completo (usar **-N** para não escapar espaços em branco e caracteres especiais):

```
tree -i -f
```

- Exibe o tamanho de cada arquivo e o tamanho acumulado de cada diretório, em um formato de leitura para humanos:

```
tree -s -h --du
```

- Exibe arquivos em uma árvore hierárquica, utilizando um padrão coringa, e eliminando diretórios que não contêm arquivos correspondentes ao informado:

```
tree -P '{{*.txt}}' --prune
```

- Exibe diretórios em uma árvore hierárquica, utilizando um padrão coringa, e eliminando diretórios que não possuem ancestrais do informado:

```
tree -P {{nome_diretorio}} --matchdirs --prune
```

- Exibe a árvore ignorando os diretórios informados:

```
tree -I '{{nome_diretorio1|nome_diretorio2}}'
```

# uname

Exibe detalhes sobre a máquina atual e o sistema operacional em execução nela.

Veja também **lsb\_release**.

Mais informações: <https://www.gnu.org/software/coreutils/uname>.

- Exibe o nome do kernel:

```
uname
```

- Exibe informações sobre a arquitetura e o processador:

```
uname --machine --processor
```

- Exibe nome do kernel, lançamento do kernel e versão do kernel:

```
uname --kernel-name --kernel-release --kernel-version
```

- Exibe o nome de rede do computador:

```
uname --nodename
```

- Exibe todas as informações disponíveis sobre o sistema:

```
uname --all
```

# units

Realiza a conversão entre duas unidades de medida.

Mais informações: <https://www.gnu.org/software/units/>.

- Rodar no modo interativo:

```
units
```

- Mostrar a conversão entre duas unidades simples:

```
units {{quarts}} {{tablespoons}}
```

- Converter entre unidades com quantidades definidas:

```
units {{15 pounds}} {{kilograms}}
```

- Mostrar a conversão entre duas unidades compostas:

```
units "{{meters / second}}" "{{inches / hour}}"
```

- Mostrar a conversão entre unidades de diferentes dimensões:

```
units "{{acres}}" "{{ft^2}}"
```

# unlzma

Este comando é um pseudônimo de **xz**.

Mais informações: <https://manned.org/unlzma>.

- Ver documentação sobre o comando original:

```
tldr xz
```

# unrar

Descompactar arquivos comprimidos no formato RAR.

Mais informações: <https://manned.org/unrar>.

- Descompactar o arquivo mantendo a estrutura de diretórios original:

```
unrar x {{arquivo.rar}}
```

- Descompactar o arquivo sem manter a estrutura de diretórios original:

```
unrar e {{arquivo.rar}}
```

- Verificar a integridade do conteúdo de um arquivo:

```
unrar t {{arquivo.rar}}
```

- Exibir o conteúdo de um arquivo sem descompactá-lo:

```
unrar l {{arquivo.rar}}
```

# unxz

Este comando é um pseudônimo de **xz**.

Mais informações: <https://manned.org/unxz>.

- Ver documentação sobre o comando original:

```
tldr xz
```

# unzip

Ferramenta de descompactação de arquivos zip.

Mais informações: <https://manned.org/unzip>.

- Extraíndo arquivos zip:

```
unzip {{arquivo.zip}}
```

- Extraíndo arquivos zip para caminhos específicos:

```
unzip {{arquivo.zip}} -d {{caminho/para}}
```

- Listando conteúdos de arquivos zip:

```
unzip -l {{arquivo.zip}}
```

- Extraíndo arquivos zip sobrescrevendo outros arquivos:

```
unzip -o {{arquivo.zip}}
```

- Extraíndo arquivos zip não sobrescrevendo outros arquivos:

```
unzip -n {{arquivo.zip}}
```

- Extraíndo arquivos zip sem a estrutura dos diretórios:

```
unzip -j {{arquivo.zip}}
```

# vi

Este comando é um pseudônimo de **vim**.

- Ver documentação sobre o comando original:

```
tldr vim
```



# vim

Vim (Vi IMproved), é um editor de texto em linha de comando, que fornece muitos modos para diferentes tipos de manipulação de texto.

Apertando **i** no modo normal entra em modo insert (inserir). Apertando **<Esc>** volta para o modo normal, que permite o uso dos comandos do Vim.

Mais informações: <https://www.vim.org>.

- Abre um arquivo:

```
vim {{caminho/para/arquivo}}
```

- Abre um arquivo em um número da linha específica:

```
vim +{{número_da_linha}} {{caminho/para/arquivo}}
```

- Abre o manual do Vim em visualização:

```
:help<Enter>
```

- Salva e sai do arquivo atual:

```
:wq<Enter>
```

- Entra em modo normal e desfaz a última operação:

```
<ESC>u
```

- Procura por um sequência padrão dentro de um arquivo (aperte n/N para ir para próxima/anterior sequência padrão):

```
/{{sequência_padrão_procurada}}<Enter>
```

- Executa uma substituição por expressão regular no arquivo todo:

```
:%s/{{expressão_regular}}/{{substituição}}/g<Enter>
```

- Mostra os números das linhas:

```
:set nu<Enter>
```

# virsh-connect

Conectar-se a um hipervisor de máquina virtual.

Veja também: **virsh**.

Mais informações: <https://manned.org/virsh>.

- Conectar-se ao hipervisor padrão:

```
virsh connect
```

- Conectar-se como root ao hipervisor local QEMU/KVM:

```
virsh connect qemu:///system
```

- Iniciar uma nova instância do hipervisor e conectar-se a ela como usuário local:

```
virsh connect qemu:///session
```

- Conectar-se como root a um hipervisor remoto usando ssh:

```
virsh connect qemu+ssh://{nome_do_usuario@nome_do_host}/  
system
```

# virsh-domblklist

Listar informações sobre dispositivos de bloco associados a uma máquina virtual.

Veja também: **virsh**.

Mais informações: <https://manned.org/virsh>.

- Listar o nome do destino e o caminho da origem dos dispositivos de bloco:

```
virsh domblklist --domain {{nome_da_vm}}
```

- Listar o tipo de disco e o valor do dispositivo, bem como o nome do destino e o caminho da origem:

```
virsh domblklist --domain {{nome_da_vm}} --details
```

# virsh-help

Exibir informações sobre comandos ou grupos de comandos do **virsh**.

Veja também: **virsh**.

Mais informações: <https://manned.org/virsh>.

- Listar os comandos do **virsh** agrupados em categorias relacionadas:

```
virsh help
```

- Listar as categorias de comandos:

```
virsh help | grep "palavra-chave"
```

- Listar os comandos de uma categoria:

```
virsh help {{palavra-chave_da_categoria}}
```

- Mostrar ajuda para um comando:

```
virsh help {{comando}}
```

# virsh-list

Liste o ID, nome e estado das máquinas virtuais.

Veja também: **virsh**.

Mais informações: <https://manned.org/virsh>.

- Listar informações sobre máquinas virtuais em execução:

```
virsh list
```

- Listar informações sobre máquinas virtuais independentemente do estado:

```
virsh list --all
```

- Listar informações sobre máquinas virtuais com autostart ativado ou desativado:

```
virsh list --all --{{autostart|no-autostart}}
```

- Listar informações sobre máquinas virtuais com ou sem snapshots:

```
virsh list --all --{{with-snapshot|without-snapshot}}
```

# virsh pool-autostart

Habilita ou desabilita a inicialização automática para um pool de armazenamento de máquina virtual.

Veja também: **virsh**.

Mais informações: <https://manned.org/virsh>.

- Habilita a inicialização automática para o pool de armazenamento especificado pelo nome ou UUID (determinado usando `virsh pool-list`):

```
virsh pool-autostart --pool {{nome|uuid}}
```

- Desabilita a inicialização automática para o pool de armazenamento especificado pelo nome ou UUID:

```
virsh pool-autostart --pool {{nome|uuid}} --disable
```

# virsh pool-build

Constrói o sistema de armazenamento subjacente para um pool de armazenamento de máquina virtual, conforme definido em seu arquivo de configuração em **/etc/libvirt/storage**.

Veja também: **virsh**, **virsh-pool-define-as**, **virsh-pool-start**.

Mais informações: <https://manned.org/virsh>.

- Constrói o pool de armazenamento especificado pelo nome ou UUID (determinado usando `virsh pool-list`):

```
virsh pool-build --pool {{nome|uuid}}
```

# virsh pool-define-as

Cria um arquivo de configuração em `/etc/libvirt/storage` para um pool de armazenamento persistente de máquina virtual a partir dos argumentos fornecidos.

Veja também: **virsh**, **virsh-pool-build**, **virsh-pool-start**.

Mais informações: <https://manned.org/virsh>.

- Criar o arquivo de configuração para um pool de armazenamento chamado `pool_name` usando `/var/vms` como o sistema de armazenamento subjacente:

```
virsh pool-define-as --name {{nome_do_pool}} --type {{dir}}  
--target {{/var/vms}}
```



# virsh pool-delete

Exclui o sistema de armazenamento subjacente de um pool de armazenamento de máquina virtual inativo.

Veja também: **virsh**, **virsh-pool-destroy**, **virsh-pool-undefine**.

Mais informações: <https://manned.org/virsh>.

- Exclui o sistema de armazenamento subjacente para o pool de armazenamento especificado pelo nome ou UUID (determinado usando `virsh pool-list`):

```
virsh pool-delete --pool {{nome|uuid}}
```

# virsh pool-destroy

Interrompe um pool de armazenamento ativo de máquina virtual.

Veja também: **virsh**, **virsh-pool-delete**.

Mais informações: <https://manned.org/virsh>.

- Interrompe um pool de armazenamento especificado pelo nome ou UUID (determinado usando `virsh pool-list`):

```
virsh pool-destroy --pool {{nome|uuid}}
```

# virsh pool-info

Lista informações sobre um pool de armazenamento de máquina virtual.

Veja também: **virsh**.

Mais informações: <https://manned.org/virsh>.

- Lista o nome, UUID, estado, tipo de persistência, status de inicialização automática, capacidade, espaço alocado e espaço disponível para o pool de armazenamento especificado pelo nome ou UUID (determinado usando `virsh pool-list`):

```
virsh pool-info --pool {{nome|uuid}}
```

# virsh pool-list

Lista informações sobre pools de armazenamento de máquinas virtuais.

Veja também: **virsh**, **virsh-pool-autostart**, **virsh-pool-define-as**.

Mais informações: <https://manned.org/virsh>.

- Lista o nome, estado e se a inicialização automática está habilitada ou desabilitada para pools de armazenamento ativos:

```
virsh pool-list
```

- Lista informações para pools de armazenamento ativos e inativos ou apenas inativos:

```
virsh pool-list --{{all|inactive}}
```

- Lista informações estendidas sobre persistência, capacidade, alocação e espaço disponível para pools de armazenamento ativos:

```
virsh pool-list --details
```

- Lista informações para pools de armazenamento ativos com inicialização automática habilitada ou desabilitada:

```
virsh pool-list --{{autostart|no-autostart}}
```

- Lista informações para pools de armazenamento ativos que são persistentes ou transitórios:

```
virsh pool-list --{{persistent|transient}}
```

- Lista o nome e UUID dos pools de armazenamento ativos:

```
virsh pool-list --name --uuid
```

# virsh pool-start

Inicia um pool de armazenamento de máquina virtual previamente configurado, mas inativo.

Veja também: **virsh**, **virsh-pool-define-as**, **virsh-pool-destroy**.

Mais informações: <https://manned.org/virsh>.

- Inicia o pool de armazenamento especificado pelo nome ou UUID (determinado usando `virsh pool-list`) e cria o sistema de armazenamento subjacente se ele não existir:

```
virsh pool-start --pool {{nome|uuid}} --build
```

# virsh pool-undefine

Exclui o arquivo de configuração em `/etc/libvirt/storage` para um pool de armazenamento de máquina virtual parado.

Veja também: **virsh**, **virsh-pool-destroy**.

Mais informações: <https://manned.org/virsh>.

- Excluir a configuração do pool de armazenamento pelo nome ou UUID especificado (determinado usando `virsh pool-list`):

```
virsh pool-undefine --pool {{nome|uuid}}
```

# virsh-undefine

Excluir uma máquina virtual.

Mais informações: <https://manned.org/virsh>.

- Excluir apenas o arquivo de configuração da máquina virtual:

```
virsh undefine --domain {{nome_da_vm}}
```

- Excluir o arquivo de configuração e todos os volumes de armazenamento associados:

```
virsh undefine --domain {{nome_da_vm}} --remove-all-storage
```

- Excluir o arquivo de configuração e os volumes de armazenamento especificados usando o nome de destino ou o nome de origem (obtido a partir do comando `virsh domblklist`):

```
virsh undefine --domain {{nome_da_vm}} --storage  
{{sda,caminho/para/origem}}
```

# virsh

Gerenciar domínios de convidados do virsh. (NOTA: 'guest\_id' pode ser o ID, nome ou UUID do convidado).

Alguns subcomandos, como **virsh list**, têm sua própria documentação de uso.

Mais informações: <https://libvirt.org/virshcmdref.html>.

- Conectar-se a uma sessão do hipervisor:

```
virsh connect {{qemu:///system}}
```

- Listar todos os domínios:

```
virsh list --all
```

- Despejar arquivo de configuração do convidado:

```
virsh dumpxml {{guest_id}} > {{caminho/para/convidado.xml}}
```

- Criar um convidado a partir de um arquivo de configuração:

```
virsh create {{caminho/para/arquivo_de_configuracao.xml}}
```

- Editar o arquivo de configuração de um convidado (o editor pode ser alterado com \$EDITOR):

```
virsh edit {{guest_id}}
```

- Iniciar/reiniciar/desligar/suspender/resumir um convidado:

```
virsh {{comando}} {{guest_id}}
```

- Salvar o estado atual de um convidado em um arquivo:

```
virsh save {{guest_id}} {{nome_do_arquivo}}
```

- Excluir um convidado em execução:

```
virsh destroy {{guest_id}} && virsh undefine {{guest_id}}
```



# vlc

Reprodutor multimídia multi-plataforma.

Mais informações: [https://wiki.videolan.org/Documentation:Command\\_line/](https://wiki.videolan.org/Documentation:Command_line/).

- Reproduzir um arquivo:

```
vlc {{caminho/para/arquivo}}
```

- Reproduzir em tela cheia:

```
vlc --fullscreen {{caminho/para/arquivo}}
```

- Reproduzir mudo:

```
vlc --no-audio {{caminho/para/arquivo}}
```

- Reproduzir repetidamente:

```
vlc --loop {{caminho/para/arquivo}}
```

- Reproduzir vídeo de um URL:

```
vlc {{https://www.youtube.com/watch?v=fxNlpQYRz7s}}
```

# wget

Baixar arquivos da Internet.

Suporta HTTP, HTTPS, e FTP.

Mais informações: <https://www.gnu.org/software/wget>.

- Baixa o conteúdo de uma URL para o arquivo (nomeado como "foo" neste caso):

```
wget {{https://exemplo.com/foo}}
```

- Baixa o conteúdo de uma URL para o arquivo (nomeado como "bar" neste caso):

```
wget --output-document {{bar}} {{https://exemplo.com/foo}}
```

- Baixa uma única página web e todo os seus recursos com intervalos de 3 segundos entre requisições (scripts, stylesheets, imagens, etc.):

```
wget --page-requisites --convert-links --wait={{3}}  
{{https://exemplo.com/algumapagina.html}}
```

- Baixa todos os arquivos listados dentro de um diretório e seus sub-diretórios (não baixa elementos de página incorporados):

```
wget --mirror --no-parent {{https://exemplo.com/  
algumcaminho/}}
```

- Limita a velocidade de download e o número de novas tentativas de conexão:

```
wget --limit-rate={{300k}} --tries={{100}} {{https://  
exemplo.com/algumcaminho/}}
```

- Baixa um arquivo de um servidor HTTP usando Autenticação Básica (também funciona para FTP):

```
wget --user={{nomeusuario}} --password={{senha}} {{https://  
exemplo.com}}
```

- Continua um download incompleto:

```
wget --continue {{https://exemplo.com}}
```

- Baixa todas as URLs armazenadas em um arquivo de texto para um diretório específico:

```
wget --directory-prefix {{caminho/para/diretorio}} --input-  
file {{URLs.txt}}
```

# where

Reporta todas as instâncias conhecidas do comando.

Pode ser um executável na variável PATH, um alias, ou um comando builtin do shell.

Mais informações: <https://zsh.sourceforge.io/Doc/Release/Shell-Builtin-Commands.html>.

- Encontra todas as instâncias de um comando:

```
where {{comando}}
```

# which

Localiza um programa no path do usuário.

Mais informações: <https://manned.org/which>.

- Procura na variável PATH e exibe a localização do executável se encontrado:

```
which {{executável}}
```

- Se existe múltiplos executáveis que concidem com a pesquisa, exibe todos:

```
which -a {{executável}}
```

# while

Loop simples da shell.

Mais informações: [https://pubs.opengroup.org/onlinepubs/9699919799/utilities/V3\\_chap02.html#tag\\_18\\_09\\_04\\_09](https://pubs.opengroup.org/onlinepubs/9699919799/utilities/V3_chap02.html#tag_18_09_04_09).

- Lê a entrada default (`stdin`) e realiza uma ação a cada linha:

```
while read line; do echo "$line"; done
```

- Executa um comando para sempre a cada segundo:

```
while :; do {{comando}}; sleep 1; done
```

# whoami

Imprime o nome do usuário associado com o ID do usuário efetivamente atual.

Mais informações: <https://www.gnu.org/software/coreutils/whoami>.

- Exibe o nome do usuário logado:

```
whoami
```

- Exibe o nome do usuário depois de uma mudança do ID do usuário:

```
sudo whoami
```

# wordgrinder

Processador de texto em linha de comando.

Mais informações: <https://cowlark.com/wordgrinder>.

- Iniciar o wordgrinder (carrega um documento vazio por padrão):

```
wordgrinder
```

- Abrir um arquivo específico:

```
wordgrinder {{nome_do_arquivo}}
```

- Mostrar o menu:

```
Alt + M
```



# wpa\_supplicant

Gerenciador de redes wireless protegidas.

Mais informações: [https://manned.org/wpa\\_supplicant.1](https://manned.org/wpa_supplicant.1).

- Entrar em uma rede wireless protegida:

```
wpa_supplicant -i {{interface}} -c {{caminho/para/  
wpa_supplicant_conf.conf}}
```

- Entrar em uma rede wireless protegida e executar o wpa\_cli em um daemon:

```
wpa_supplicant -B -i {{interface}} -c {{caminho/para/  
wpa_supplicant_conf.conf}}
```

# x11docker

Executar aplicativos de GUI e interfaces de desktop seguramente em contêineres do Docker.

Veja também **xephyr**.

Mais informações: <https://github.com/mviereck/x11docker>.

- Iniciar o VLC em um contêiner:

```
x11docker --pulseaudio --share={{$HOME/Videos}} {{jess/vlc}}
```

- Iniciar o Xfce em uma janela:

```
x11docker --desktop {{x11docker/xfce}}
```

- Iniciar o GNOME em uma janela:

```
x11docker --desktop --gpu --init={{systemd}} {{x11docker/gnome}}
```

- Iniciar o KDE Plasma em uma janela:

```
x11docker --desktop --gpu --init={{systemd}} {{x11docker/kde-plasma}}
```

- Exibir ajuda:

```
x11docker --help
```

# XCV

Corta, copia e cola na linha de comando.

Mais informações: <https://github.com/busterc/xcv>.

- Cortar um arquivo:

```
xcv x {{arquivo_a_ser_cortado}}
```

- Copiar um arquivo:

```
xcv c {{arquivo_a_ser_copiado}}
```

- Colar um arquivo:

```
xcv v {{arquivo_a_ser_colado}}
```

- Listar todos os arquivos disponíveis para serem colados:

```
xcv l
```

# xkill

Termina o cliente associado a um elemento gráfico.

Utilizado para forçar a terminação de processos que não respondem ou não apresentam botão "fechar".

Mais informações: <https://www.x.org/releases/current/doc/man/man1/xkill.1.xhtml>.

- Ativar um cursor para fechar uma janela com o clique do botão esquerdo do mouse (pressionar qualquer outro botão para cancelar):

```
xkill
```

# XZ

Compactar ou descompactar arquivos com a extensão .xz ou .lzma.

Mais informações: <https://manned.org/xz>.

- Compactar um arquivo no formato xz:

```
xz {{arquivo}}
```

- Descompactar um arquivo no formato xz:

```
xz -d {{arquivo.xz}}
```

- Compactar um arquivo no formato lzma:

```
xz --format=lzma {{arquivo}}
```

- Descompactar um arquivo no formato lzma:

```
xz -d --format=lzma {{arquivo.lzma}}
```

- Descompactar um arquivo e escrever a saída no terminal:

```
xz -dc {{arquivo.xz}}
```

- Compactar um arquivo sem apagar o arquivo original:

```
xz -k {{arquivo}}
```

- Compactar um arquivo utilizando a compactação mais rápida:

```
xz -0 {{arquivo}}
```

- Compactar um arquivo utilizando a compactação mais eficiente:

```
xz -9 {{arquivo}}
```

# xzcat

Este comando é um pseudônimo de **xz**.

Mais informações: <https://manned.org/xzcat>.

- Ver documentação sobre o comando original:

```
tldr xz
```

# yarn-why

Identifica por que um pacote Yarn foi instalado.

Mais informações: <https://github.com/amio/yarn-why>.

- Exibir na tela o motivo de um pacote Yarn estar instalado:

```
yarn-why {{nome_do_pacote}}
```

# yes

Exibe algo na tela repetidamente.

Mais informações: <https://www.gnu.org/software/coreutils/yes>.

- Exibir na tela a palavra "mensagem" repetidamente:

```
yes {{mensagem}}
```

- Exibir na tela a letra "y" repetidamente:

```
yes
```



# zip

Ferramenta de compressão de arquivos em arquivos zip.

Mais informações: <https://manned.org/zip>.

- Compactando arquivos em um arquivo zip:

```
zip {{output.zip}} {{arquivo1}} {{arquivo2}} {{arquivo3}}
```

- Compactando todos os arquivos de um diretório:

```
zip {{output.zip}} {{caminho/do/diretorio/*}}
```

- Adicionando arquivos a um arquivo zip existente:

```
zip {{arquivo_existente.zip}} {{caminho/do/diretorio}}
```

- Compactando todos os arquivos de um diretório mantendo estruturas de diretórios:

```
zip -r {{output.zip}} {{caminho/do/diretorio}}
```

- Compactando arquivos de um diretório excluindo arquivos específicos:

```
zip -r {{output.zip}} {{caminho/do/diretorio}} -x {{caminho/a/ser/excluido}}
```

- Compactando arquivos definindo o nível de compressão [9]:

```
zip -r -{{9}} {{output.zip}} {{caminho/do/diretorio}}
```

- Deletando arquivos de um arquivo zip:

```
zip -d {{output.zip}} "{{foo/*.ext}}"
```

# zsh

Z SHell, um interpretador de linha de comando compatível com o Bash.

Veja também **bash**, **histexpand**.

Mais informações: <https://www.zsh.org>.

- Inicie uma sessão shell interativa:

```
zsh
```

- Execute [c]omandos específicos:

```
zsh -c "{{echo Olá Mundo}}"
```

- Execute um script específico:

```
zsh {{caminho/para/script.zsh}}
```

- Verifica um script específico por erros de sintaxe sem executá-lo:

```
zsh --no-exec {{caminho/para/script.zsh}}
```

- Executa comandos específicos da `stdin`:

```
{{echo Olá Mundo}} | zsh
```

- Execute um script específico, imprimindo cada comando do script antes de executá-lo:

```
zsh --xtrace {{caminho/para/script.zsh}}
```

- Inicie uma sessão shell interativa no modo verboso, imprimindo cada comando antes de executá-lo:

```
zsh --verbose
```

- Executa um comando específico dentro do `zsh` com padrões glob desativados:

```
noglob {{comando}}
```

Linux

# a2disconf

Desativar um arquivo de configuração em sistemas operacionais baseados no Debian.

Mais informações: <https://manpages.debian.org/latest/apache2/a2disconf.8.en.html>.

- Desativa um arquivo de configuração:

```
sudo a2disconf {{arquivo_de_configuracao}}
```

- Não mostra mensagens informativas:

```
sudo a2disconf --quiet {{arquivo_de_configuracao}}
```

# a2dismod

Desativa um módulo do Apache em sistemas operacionais baseados no Debian.

Mais informações: <https://manpages.debian.org/latest/apache2/a2dismod.8.en.html>.

- Desativa um módulo:

```
sudo a2dismod {{módulo}}
```

- Não mostra mensagens informativas:

```
sudo a2dismod --quiet {{módulo}}
```

# a2dissite

Desativa um host virtual do Apache em sistemas operacionais baseados no Debian.

Mais informações: <https://manpages.debian.org/latest/apache2/a2dissite.8.en.html>.

- Desativa um host virtual:

```
sudo a2dissite {{host_virtual}}
```

- Não mostra mensagens informativas:

```
sudo a2dissite --quiet {{host_virtual}}
```

# a2enconf

Ativa um arquivo de configuração do Apache em sistemas operacionais baseados no Debian.

Mais informações: <https://manpages.debian.org/latest/apache2/a2enconf.8.en.html>.

- Ativa um arquivo de configuração:

```
sudo a2enconf {{arquivo_de_configuração}}
```

- Não mostra mensagens informativas:

```
sudo a2enconf --quiet {{arquivo_de_configuração}}
```

# a2enmod

Ativa um módulo do Apache em sistemas operacionais baseados no Debian.

Mais informações: <https://manpages.debian.org/latest/apache2/a2enmod.8.en.html>.

- Ativa um módulo:

```
sudo a2enmod {{módulo}}
```

- Não mostra mensagens informativas:

```
sudo a2enmod --quiet {{module}}
```



# a2ensite

Ativa um host virtual do Apache em sistemas operacionais baseados no Debian.

Mais informações: <https://manpages.debian.org/latest/apache2/a2ensite.8.en.html>.

- Ativa um host virtual:

```
sudo a2ensite {{host_virtual}}
```

- Não mostra mensagens informativas:

```
sudo a2ensite --quiet {{host_virtual}}
```

# a2query

Exibe configurações de execução do Apache em sistemas operacionais baseados no Debian.

Mais informações: <https://manpages.debian.org/latest/apache2/a2query.html>.

- Lista módulos ativos do Apache:

```
sudo a2query -m
```

- Verifica se um módulo específico está instalado:

```
sudo a2query -m {{nome_do_modulo}}
```

- Lista host virtuais ativos:

```
sudo a2query -s
```

- Exibe o módulo de multi processamento atualmente ativo:

```
sudo a2query -M
```

- Mostra a versão do Apache:

```
sudo a2query -v
```

# abbr

Gerencie abreviações para fish-shell.

Palavras definidas pelo usuário são substituídas por frases longas assim que são digitadas.

Mais informações: <https://fishshell.com/docs/current/cmds/abbr.html>.

- Adicione uma nova abreviação:

```
abbr --add {{nome_abreviacao}} {{comando}}  
{{argumentos_comando}}
```

- Renomear uma abreviação existente:

```
abbr --rename {{nome_antigo}} {{novo_nome}}
```

- Apagar uma abreviação existente:

```
abbr --erase {{nome_abreviacao}}
```

- Importar abreviações definidas em outro host via SSH:

```
ssh {{nome_host}} abbr --show | source
```

# abroot

O utilitário ABRoot fornece total imutabilidade e atomicidade ao transacionar entre 2 estados da partição raiz ( $A \rightleftharpoons B$ ).

Isso também permite transações sob demanda via um shell transacional.

Mais informações: <https://github.com/Vanilla-OS/ABRoot>.

- Saída do estado da partição raiz atual ou futuro:

```
sudo abroot get {{present|future}}
```

- Insira um shell transacional na futura partição raiz e alterne o root na próxima inicialização:

```
sudo abroot shell
```

- Executa um comando específico no shell transacional na partição raiz futura e troca para ela na próxima inicialização:

```
sudo abroot exec "{{comando}}"
```

- Instala pacotes específicos no servidor dentro do shell transacional na partição raiz futura e troca para ela na próxima inicialização:

```
sudo abroot exec apt install {{pacote1 pacote2 ...}}
```

- Atualiza a partição de inicialização (apenas para usuários avançados):

```
sudo abroot _update-boot
```

- Exibe ajuda:

```
abroot --help
```

- Exibe versão:

```
abroot --version
```

# ac

Imprime estatísticas de quanto tempo usuários permaneceram conectados.

Mais informações: <https://www.gnu.org/software/acct/manual/accounting.html#ac>.

- Imprime quanto tempo em horas o usuário atual ficou conectado:

```
ac
```

- Imprime quanto tempo em horas usuários ficaram conectados:

```
ac --individual-totals
```

- Imprime quanto tempo em horas um usuário em particular ficou conectado:

```
ac --individual-totals {{usuario}}
```

- Imprime quanto tempo um usuário em particular ficou conectado em horas por dia (com total):

```
ac --daily-totals --individual-totals {{usuario}}
```

- Também exibe detalhes adicionais:

```
ac --compatibility
```

# acountry

Imprime o país onde um endereço IPv4 ou nome do servidor estão localizados.

Mais informações: <https://manned.org/acountry>.

- Imprime um país onde um endereço IPv4 ou host está localizado:

```
acountry {{exemplo.com}}
```

- Imprime uma saída de [d]epuração extra:

```
acountry -d {{exemplo.com}}
```

- Imprime informações mais [v]erbosas:

```
acountry -v {{exemplo.com}}
```

# acpi

Exibe status da bateria ou informações térmicas.

Mais informações: <https://sourceforge.net/projects/acpiclient/files/acpiclient/>.

- Exibe informações sobre a bateria:

```
acpi
```

- Exibe informações térmicas:

```
acpi -t
```

- Exibe informações sobre dispositivo de resfriamento:

```
acpi -c
```

- Exibe informações térmicas em Fahrenheit:

```
acpi -tf
```

- Exibe todas as informações:

```
acpi -V
```

- Extrai informações de `/proc` em vez de `/sys`:

```
acpi -p
```

# add-apt-repository

Gerenciar definições de repositórios APT.

Mais informações: <https://manned.org/apt-add-repository>.

- Adicionar um repositório:

```
add-apt-repository {{especificacao_do_repositorio}}
```

- Remover um repositório:

```
add-apt-repository --remove {{especificacao_do_repositorio}}
```

- Adicionar um repositório e atualizar o cache do(s) pacote(s) deste repositório:

```
add-apt-repository --update {{especificacao_do_repositorio}}
```

- Adicionar um repositório e habilitar o download do código fonte do(s) pacote(s) deste repositório:

```
add-apt-repository --enable-source  
{{especificacao_do_repositorio}}
```



# addpart

Informa ao kernel do Linux sobre a existência da partição especificada.

O comando é um wrapper do ioctl **add partition**.

Mais informações: <https://manned.org/addpart>.

- Informar ao kernel do Linux sobre a existência da partição especificada:

```
addpart {{dispositivo}} {{particao}} {{inicio}} {{tamanho}}
```

# addr2line

Converte endereços de um binário em nomes de arquivos e números de linha.

Mais informações: <https://manned.org/addr2line>.

- Exibe o nome do arquivo e o número da linha do código-fonte de um endereço de instrução de um executável:

```
addr2line --exe={{caminho/do/executavel}} {{endereco}}
```

- Exibe o nome da função, nome do arquivo e número da linha:

```
addr2line --exe={{caminho/do/executavel}} --functions  
{{endereco}}
```

- Desembaraça o nome da função em código C++:

```
addr2line --exe={{caminho/do/executavel}} --functions --  
demangle {{endereco}}
```

# adduser

Utilitário para criação de novos usuários.

Mais informações: <https://manpages.debian.org/latest/adduser/adduser.html>.

- Criar um novo usuário, o seu diretório na pasta home e solicitar o preenchimento da sua senha:

```
adduser {{nome_do_usuario}}
```

- Criar um novo usuário sem o seu diretório na pasta home:

```
adduser --no-create-home {{nome_do_usuario}}
```

- Criar um novo usuário especificando a localização do seu diretório:

```
adduser --home {{caminho_da_pasta_do_usuario}}  
{{nome_do_usuario}}
```

- Criar um novo usuário e configurar o seu shell de login:

```
adduser --shell {{caminho_para_o_shell}} {{nome_do_usuario}}
```

- Criar um novo usuário e atribuí-lo a um grupo:

```
adduser --ingroup {{grupo}} {{nome_do_usuario}}
```

# adig

Imprime informações recebidas dos servidores do Sistema de Domínio de Nome(DNS).

Mais informações: <https://manned.org/adig>.

- Exibe uma gravação A (padrão) do DNS por nome(s) de servidor(es):

```
adig {{exemplo.com}}
```

- Exibe uma saída de [d]epuração extra:

```
adig -d {{exemplo.com}}
```

- Conecte-se a um servidor DNS específico:

```
adig -s {{1.2.3.4}} {{exemplo.com}}
```

- Use uma porta TCP específica para se conectar ao servidor DNS:

```
adig -T {{port}} {{exemplo.com}}
```

- Use uma porta UDP específica para se conectar ao servidor DNS:

```
adig -U {{port}} {{exemplo.com}}
```

# alien

Converter diferentes pacotes de instalação para outros formatos.

Mais informações: <https://manned.org/alien>.

- Converter um arquivo de instalação específico para o formato Debian (extensão `.deb`):

```
sudo alien --to-deb {{caminho/para/arquivo}}
```

- Converter um arquivo de instalação específico para o formato Red Hat (extensão `.rpm`):

```
sudo alien --to-rpm {{caminho/para/arquivo}}
```

- Converter um arquivo de instalação específico para um arquivo de instalação do Slackware (extensão `.tgz`):

```
sudo alien --to-tgz {{caminho/para/arquivo}}
```

- Converter um arquivo de instalação específico para o formato Debian e instalar no sistema:

```
sudo alien --to-deb --install {{caminho/para/arquivo}}
```

# alpine

Um cliente de e-mail e programa de newsgroup Usenet com uma interface inspirada no pico e nano.

Suporta a maioria dos serviços de e-mail por meio de IMAP.

Mais informações: <https://manned.org/alpine>.

- Inicia o alpine:

```
alpine
```

- Abre o alpine na tela de composição de mensagem com o e-mail do destinatário preenchido:

```
alpine {{email@exemplo.net}}
```

- Encerra o alpine:

```
'q' e 'y'
```

# alternatives

Este comando é um pseudônimo de **update-alternatives**.

Mais informações: <https://manned.org/alternatives>.

- Ver documentação sobre o comando original:

```
tldr update-alternatives
```

# apache2ctl

Interface de controle do servidor web HTTP Apache.

Este comando está disponível nas distribuições baseadas em Debian, para as baseadas em RHEL veja **httpd**.

Mais informações: <https://manpages.debian.org/latest/apache2/apache2ctl.8.en.html>.

- Iniciar o Apache. Caso ele já esteja em execução, uma mensagem será apresentada:

```
sudo apache2ctl start
```

- Encerrar o Apache:

```
sudo apache2ctl stop
```

- Reiniciar o Apache:

```
sudo apache2ctl restart
```

- Verificar se o arquivo de configuração está correto sintaticamente:

```
sudo apache2ctl -t
```

- Listar os módulos carregados:

```
sudo apache2ctl -M
```



# apk

Gerenciador de pacotes da distribuição Alpine.

Mais informações: [https://wiki.alpinelinux.org/wiki/Alpine\\_Linux\\_package\\_management](https://wiki.alpinelinux.org/wiki/Alpine_Linux_package_management).

- Atualizar os índices dos pacotes disponíveis:

```
apk update
```

- Instalar um pacote:

```
apk add {{pacote}}
```

- Remover um pacote:

```
apk del {{pacote}}
```

- Reparar ou atualizar um pacote sem modificar as principais dependências:

```
apk fix {{pacote}}
```

- Procurar um pacote especificando alguma palavra-chave:

```
apk search {{palavra_chave}}
```

- Exibir informações sobre um pacote:

```
apk info {{pacote}}
```

# apt-cache

Buscador de pacotes para distribuições baseadas no Debian.

Mais informações: <https://manpages.debian.org/latest/apt/apt-cache.8.html>.

- Buscar pacotes, no cache de pacotes APT, correspondentes ao critério de busca:

```
apt-cache search {{critério_de_busca}}
```

- Exibir informações sobre um pacote:

```
apt-cache show {{nome_do_pacote}}
```

- Informar a situação de um pacote, se ele está instalado e atualizado:

```
apt-cache policy {{nome_do_pacote}}
```

- Exibir as dependências de um pacote:

```
apt-cache depends {{nome_do_pacote}}
```

- Exibir pacotes dependentes de um determinado pacote:

```
apt-cache rdepends {{nome_do_pacote}}
```

# apt-file

Buscador de arquivos nos pacotes apt, incluindo os não instalados.

Mais informações: <https://manpages.debian.org/latest/apt-file/apt-file.1.html>.

- Atualizar as informações dos pacotes a partir de todos os repositórios remotos:

```
sudo apt update
```

- Buscar por pacotes que contêm o arquivo ou caminho especificado:

```
apt-file search {{nome_do_pacote_ou_caminho}}
```

- Listar o conteúdo de um pacote específico:

```
apt-file list {{nome_do_pacote}}
```

# apt-get

Gerenciador de pacotes das distribuições baseadas em Debian.

Procure por pacotes utilizando o **apt-cache**.

Mais informações: <https://manpages.debian.org/latest/apt/apt-get.8.html>.

- Atualizar a lista de pacotes disponíveis (recomenda-se executá-lo antes de outros comandos `apt-get`):

```
apt-get update
```

- Instalar um pacote ou atualizá-lo para a versão mais recente:

```
apt-get install {{nome_do_pacote}}
```

- Remover um pacote:

```
apt-get remove {{nome_do_pacote}}
```

- Remover um pacote e os seus arquivos de configuração:

```
apt-get purge {{nome_do_pacote}}
```

- Atualizar todos os pacotes instalados para as versões mais recentes:

```
apt-get upgrade
```

- Limpar o repositório local — removendo os arquivos de pacotes (`.deb`) de downloads interrompidos que não podem mais ser baixados:

```
apt-get autoclean
```

- Remover todos os pacotes obsoletos:

```
apt-get autoremove
```

- Atualizar os pacotes instalados (semelhante ao `upgrade`), porém removendo os obsoletos e instalando pacotes solicitados por novas dependências:

```
apt-get dist-upgrade
```

# apt-key

Gerenciador de chaves utilizado pelo gerenciador de pacotes APT nas distribuições baseadas em Debian.

Mais informações: <https://manpages.debian.org/latest/apt/apt-key.8.html>.

- Exibir as chaves confiáveis:

```
apt-key list
```

- Adicionar uma chave na lista de chaves confiáveis:

```
apt-key add {{arquivo_da_chave_publica.asc}}
```

- Remover uma chave da lista de chaves confiáveis:

```
apt-key del {{key_id}}
```

- Adicionar uma chave remota na lista de chaves confiáveis:

```
wget -q0 - {{https://host.tld/arquivo.key}} | apt-key add -
```

- Adicionar uma chave, de um servidor de chaves, na lista de chaves confiáveis:

```
apt-key adv --keyserver {{pgp.mit.edu}} --recv {{KEYID}}
```

# apt-mark

Utilitário que altera as configurações dos pacotes instalados.

Mais informações: <https://manpages.debian.org/latest/apt/apt-mark.8.html>.

- Marcar um pacote como instalado automaticamente:

```
sudo apt-mark auto {{nome_do_pacote}}
```

- Bloquear um pacote na sua versão atual, impedindo que ele seja atualizado:

```
sudo apt-mark hold {{nome_do_pacote}}
```

- Desbloquear um pacote, permitindo que ele seja atualizado:

```
sudo apt-mark unhold {{nome_do_pacote}}
```

- Listar os pacotes instalados manualmente:

```
apt-mark showmanual
```

- Listar os pacotes bloqueados:

```
apt-mark showhold
```

# apt

Gerenciador de pacotes das distribuições baseadas em Debian.

Mais informações: <https://manpages.debian.org/latest/apt/apt.8.html>.

- Atualizar a lista de pacotes disponíveis (recomenda-se executá-lo antes de outros comandos `apt`):

```
sudo apt update
```

- Buscar pacotes correspondentes ao critério de busca:

```
apt search {{critério_de_busca}}
```

- Exibir as informações de pacote:

```
apt show {{nome_do_pacote}}
```

- Instalar um pacote ou atualizá-lo para a versão mais recente:

```
sudo apt install {{nome_do_pacote}}
```

- Remover um pacote (Para remover os arquivos de configuração deve-se usar a opção `purge` ao invés do `remove`):

```
sudo apt remove {{nome_do_pacote}}
```

- Atualizar os pacotes instalados para as versões mais recentes:

```
sudo apt upgrade
```

# aptitude

Gerenciador de pacotes das distribuições baseadas em Debian.

Mais informações: <https://manpages.debian.org/latest/aptitude/aptitude.8.html>.

- Atualizar a lista de pacotes disponíveis (recomenda-se executá-lo antes de outros comandos `aptitude`):

```
aptitude update
```

- Instalar um novo pacote e suas dependências:

```
aptitude install {{nome_do_pacote}}
```

- Buscar pacotes correspondentes ao critério de busca:

```
aptitude search {{critério_de_busca}}
```

- Remover um pacote e todos que dependam dele:

```
aptitude remove {{nome_do_pacote}}
```

- Atualizar os pacotes instalados para as versões mais recentes:

```
aptitude upgrade
```

- Atualizar os pacotes instalados (semelhante ao `upgrade`), porém removendo os obsoletos e instalando pacotes solicitados por novas dependências:

```
aptitude full-upgrade
```



# archey

Ferramenta que exibe informações do sistema de forma estilizada.

Mais informações: <https://lclarkmichalek.github.io/archey3/>.

- Exibir as informações do sistema:

```
archey
```

# archinstall

Instalador do Arch Linux guiado com uma torção.

Mais informações: <https://archinstall.readthedocs.io>.

- Inicia o instalador interativo:

```
archinstall
```

- Inicia um instalador predefinido:

```
archinstall {{minimal|unattended}}
```

# as

Assembler GNU multiplataforma.

Seu objetivo inicial é realizar o montagem do arquivo gerado pelo **gcc** para ser utilizado pelo **ld**.

Mais informações: <https://manned.org/as>.

- Realizar a montagem de um arquivo, o resultado dessa operação será gravado no arquivo a.out:

```
as {{arquivo.s}}
```

- Realizar a montagem de um arquivo, o resultado dessa operação será gravado em um arquivo específico:

```
as {{arquivo.s}} -o {{saida.o}}
```

- Realizar a montagem de um arquivo rapidamente, pois ignora o pré-processamento de comentários e espaços em branco. (Deve ser utilizado apenas em compiladores confiáveis):

```
as -f {{arquivo.s}}
```

- Adiciona um caminho na lista de diretórios onde será realizada a busca por arquivos especificados na diretiva .include:

```
as -I {{caminho_para_o_diretorio}} {{arquivo.s}}
```

# asciiart

Converte imagens para ASCII.

Mais informações: <https://github.com/nodanaonlyzuul/asciiart>.

- Lê uma imagem de um arquivo e imprime em ASCII:

```
asciiart {{caminho/para/imagem.jpg}}
```

- Lê uma imagem de uma URL e imprime em ASCII:

```
asciiart {{www.exemplo.com/imagem.jpg}}
```

- Escolha a largura da saída (o padrão é 100):

```
asciiart --width {{50}} {{caminho/para/imagem.jpg}}
```

- Imprimir com cor:

```
asciiart --color {{caminho/para/imagem.jpg}}
```

- Escolha o formato de saída (o padrão é text):

```
asciiart --format {{text|html}} {{caminho/para/imagem.jpg}}
```

- Inverter o mapeamento dos caracteres:

```
asciiart --invert-chars {{caminho/para/imagem.jpg}}
```

# aspell

Verificador ortográfico interativo.

Mais informações: <http://aspell.net/>.

- Verificar a ortografia do texto de um arquivo:

```
aspell check {{arquivo}}
```

- Exibir as palavras escritas incorretamente no terminal:

```
cat {{arquivo}} | aspell list
```

- Exibir os dicionários disponíveis:

```
aspell dicts
```

- Executar `aspell` utilizando uma língua diferente (informe o código ISO 639 da língua):

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

- Exibir os erros ortográficos no terminal e ignorando as palavras da lista pessoal:

```
cat {{arquivo}} | aspell --personal={{lista_pessoal.pws}}  
list
```

# at

Executa comandos em um determinado momento.

Mais informações: <https://man.archlinux.org/man/at.1>.

- Iniciar o prompt `at` para que um novo conjunto de comandos seja agendado, pressione `Ctrl + D` para salvar e sair:

```
at {{hh:mm:ss}}
```

- Executar os comandos e enviar o resultado por e-mail utilizando algum programa de envio de e-mail local, por exemplo o `sendmail`:

```
at {{hh:mm:ss}} -m
```

- Executar um script em um determinado momento:

```
at {{hh:mm:ss}} -f {{caminho_para_o_script}}
```

# authconfig

Interface de linha comandos para configurar o sistema de autenticação.

Mais informações: [https://access.redhat.com/documentation/en-us/red\\_hat\\_enterprise\\_linux/7/html/system-level\\_authentication\\_guide/authconfig-install](https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/system-level_authentication_guide/authconfig-install).

- Exibir as configurações atuais (ou dry run):

```
authconfig --test
```

- Configurar o servidor para utilizar diferentes algoritmos de hash para as senhas:

```
authconfig --update --passalgo={{algoritmo}}
```

- Habilitar a autenticação via LDAP:

```
authconfig --update --enableldapauth
```

- Desabilitar a autenticação via LDAP:

```
authconfig --update --disableldapauth
```

- Habilitar o Network Information Service (NIS):

```
authconfig --update --enablenis
```

- Habilitar Kerberos:

```
authconfig --update --enablekrb5
```

- Habilitar a autenticação Winbind (Active Directory):

```
authconfig --update --enablewinbindauth
```

- Habilitar a autorização local:

```
authconfig --update --enablelocauthorize
```

# autorandr

Altera o layout da tela automaticamente.

Mais informações: <https://github.com/phillipberndt/autorandr>.

- Salvar o layout da tela em uso:

```
autorandr -s {{nome_do_perfil}}
```

- Exibir os perfis salvos:

```
autorandr
```

- Alterar o perfil:

```
autorandr -l {{nome_do_perfil}}
```

- Definir o perfil padrão:

```
autorandr -d {{nome_do_perfil}}
```



# batcat

Este comando é um pseudônimo de **bat**.

Mais informações: <https://github.com/sharkdp/bat>.

- Ver documentação sobre o comando original:

```
tldr bat
```

# battop

Um visualizador interativo para as baterias instaladas no seu notebook.

Mais informações: <https://github.com/svartalf/rust-battop>.

- Exibe informações da bateria:

```
battop
```

- Altera a [u]nidade de medida das informações da bateria (padrão: human):

```
battop -u {{human|si}}
```

# beep

Utilitário que permite o computador emitir sons.

Mais informações: <https://github.com/spkr-beep/beep>.

- Emitir um som:

```
beep
```

- Emitir um som repetidamente:

```
beep -r {{repeticoes}}
```

- Emitir um som em uma frequência (Hz) específica e com duração específica (milisegundos):

```
beep -f {{frequencia}} -l {{duracao}}
```

- Emitir cada frequência e duração como um som diferente:

```
beep -f {{frequencia}} -l {{duracao}} -n -f {{frequencia}} -l {{duracao}}
```

- Executar a escala de Dó maior:

```
beep -f {{262}} -n -f {{294}} -n -f {{330}} -n -f {{349}} -n  
-f {{392}} -n -f {{440}} -n -f {{494}} -n -f {{523}}
```

# blkid

Lista todas as partições reconhecidas e seu Identificador Único Universal (UUID).

Mais informações: <https://manned.org/blkid>.

- Lista todas as partições:

```
sudo blkid
```

- Lista todas as partições em uma tabela, incluindo os pontos de montagem atuais:

```
sudo blkid -o list
```

# bluetoothctl

Gerencia dispositivos Bluetooth a partir da linha de comando.

Mais informações: <https://bitbucket.org/serkanp/bluetoothctl>.

- Inicia o shell `bluetoothctl`:

```
bluetoothctl
```

- Lista todos os dispositivos conhecidos:

```
bluetoothctl devices
```

- Liga ou desliga o controlador Bluetooth:

```
bluetoothctl power {{on|off}}
```

- Emparelha com um dispositivo:

```
bluetoothctl pair {{endereço_mac}}
```

- Remove um dispositivo:

```
bluetoothctl remove {{endereço_mac}}
```

- Conecta a um dispositivo pareado:

```
bluetoothctl connect {{endereço_mac}}
```

- Desconecta um dispositivo pareado:

```
bluetoothctl disconnect {{endereço_mac}}
```

- Exibe ajuda:

```
bluetoothctl help
```

# bmon

Monitora a largura de banda e produz estatísticas relacionadas a rede.

Mais informações: <https://github.com/tgraf/bmon>.

- Exibir uma lista com todas as interfaces de rede:

```
bmon -a
```

- Exibir as taxas de transferência de dados em bits por segundo:

```
bmon -b
```

- Definir quais interfaces serão visíveis:

```
bmon -p {{interface_1,interface_2,interface_3}}
```

- Definir o intervalo (em segundos) que a taxa por contador será calculada:

```
bmon -R {{2.0}}
```

# bpftrace

Linguagem de análise de alto nível para eBPF Linux.

Mais informações: <https://github.com/iovisor/bpftrace>.

- Verifique a versão do bpftrace:

```
bpftrace -V
```

- Lista todos os probes:

```
sudo bpftrace -l
```

- Rode um programa de uma linha (e.g. número de syscalls por programa):

```
sudo bpftrace -e '{{tracepoint:raw_syscalls:sys_enter {  
@[comm] = count(); }}}'
```

- Rode um programa de um arquivo:

```
sudo bpftrace {{caminho/do/arquivo}}
```

- Analise um programa por PID:

```
sudo bpftrace -e '{{tracepoint:raw_syscalls:sys_enter /pid ==  
123/ { @[comm] = count(); }}}'
```

- Mostre o resultado do programa em eBPF, sem rodar ele:

```
sudo bpftrace -d -e '{{programa_de_uma_linha}}'
```

# br

Navega árvore de diretórios interativamente.

Veja também: **broot**.

Mais informações: <https://github.com/Canop/broot>.

- Começa e navega na árvore de diretórios atual interativamente:

```
br
```

- Inicia mostrando o tamanho de arquivos e diretórios:

```
br --sizes
```

- Inicia mostrando permissões:

```
br --permissions
```

- Inicia mostrando apenas diretórios:

```
br --only-folders
```

- Inicia mostrando arquivos ocultos e diretórios:

```
br --hidden
```



# brctl

Administração de pontes de rede.

Mais informações: <https://manned.org/brctl>.

- Exibir uma lista com informações das pontes de rede existentes:

```
sudo brctl show
```

- Cria uma ponte de rede:

```
sudo brctl add {{nome_da_ponte}}
```

- Remover uma ponte de rede:

```
sudo brctl del {{nome_da_ponte}}
```

- Adicionar uma interface de rede em uma ponte de rede existente:

```
sudo brctl addif {{nome_da_ponte}}  
{{nome_da_interface_de_rede}}
```

- Remover uma interface de rede de uma ponte de rede existente:

```
sudo brctl delif {{nome_da_ponte}}  
{{nome_da_interface_de_rede}}
```

# bspwm

Este comando é um pseudônimo de **bsp**pc.

Mais informações: <https://github.com/baskerville/bspwm>.

- Ver documentação sobre o comando original:

```
tldr bspc
```

# btrfs balance

Balanceia grupos de blocos em um sistema de arquivos btrfs.

Mais informações: <https://btrfs.readthedocs.io/en/latest/btrfs-balance.html>.

- Mostra o status de uma operação balance em execução ou pausada:

```
sudo btrfs balance status {{caminho/para/
sistema_de_arquivos_btrfs}}
```

- Balanceia todos os grupos de blocos (lento; reescreve todos os blocos no sistema de arquivos):

```
sudo btrfs balance start {{caminho/para/
sistema_de_arquivos_btrfs}}
```

- Balanceia grupos de blocos de dados com menos de 15% de utilização, executando a operação em segundo plano:

```
sudo btrfs balance start --bg -dusage={{15}} {{caminho/para/
sistema_de_arquivos_btrfs}}
```

- Balanceia um máximo de 10 partes de metadados com menos de 20% de utilização e pelo menos 1 parte em um determinado dispositivo `devid` (consulte `btrfs filesystem show`):

```
sudo btrfs balance start -
musage={{20}},limit={{10}},devid={{devid}} {{caminho/para/
sistema_de_arquivos_btrfs}}
```

- Converte blocos de dados para raid6 e metadados para raid1c3 (veja `mkfs.btrfs(8)` para perfis):

```
sudo btrfs balance start -dconvert={{raid6}} -
mconvert={{raid1c3}} {{caminho/para/
sistema_de_arquivos_btrfs}}
```

- Converte blocos de dados para raid1, pulando pedaços já convertidos (por exemplo, após uma operação de conversão cancelada anterior):

```
sudo btrfs balance start -dconvert={{raid1}},soft {{caminho/
para/sistema_de_arquivos_btrfs}}
```

- Cancela, pausa ou retoma uma operação de balanceamento em execução ou pausada:

```
sudo btrfs balance {{cancel|pause|resume}} {{caminho/para/  
sistema_de_arquivos_btrfs}}
```

# btrfs check

Verifica ou repara um sistema de arquivos btrfs.

Mais informações: <https://btrfs.readthedocs.io/en/latest/btrfs-check.html>.

- Verifica um sistema de arquivos btrfs:

```
sudo btrfs check {{caminho/para/partição}}
```

- Verifica e repara um sistema de arquivos btrfs (perigoso):

```
sudo btrfs check --repair {{caminho/para/partição}}
```

- Mostra o andamento da verificação:

```
sudo btrfs check --progress {{caminho/para/partição}}
```

- Verifica a soma de verificação de cada bloco de dados (se o sistema de arquivos estiver bom):

```
sudo btrfs check --check-data-csum {{caminho/para/partição}}
```

- Usa o n-ésimo superbloco (n pode ser 0, 1 ou 2):

```
sudo btrfs check --super {{n}} {{caminho/para/partição}}
```

- Reconstrói a árvore de soma de verificação:

```
sudo btrfs check --repair --init-csum-tree {{caminho/para/partição}}
```

- Reconstrói a árvore de extensão:

```
sudo btrfs check --repair --init-extent-tree {{caminho/para/partição}}
```

# btrfs device

Gerencia dispositivos em um sistema de arquivos btrfs.

Mais informações: <https://btrfs.readthedocs.io/en/latest/btrfs-device.html>.

- Adiciona um ou mais dispositivos a um sistema de arquivos btrfs:

```
sudo btrfs device add {{caminho/para/bloco_do_dispositivo1}}  
[{{caminho/para/bloco_do_dispositivo2}}] {{caminho/para/  
sistema_de_arquivos_btrfs}}
```

- Remove um dispositivo de um sistema de arquivos btrfs:

```
sudo btrfs device remove {{caminho/para/dispositivo|  
id_do_dispositivo}} [{{...}}]
```

- Exibe estatísticas de erro:

```
sudo btrfs device stats {{caminho/para/  
sistema_de_arquivos_btrfs}}
```

- Examina todos os discos e informa ao kernel todos os sistemas de arquivos btrfs detectados:

```
sudo btrfs device scan --all-devices
```

- Exibe estatísticas detalhadas de alocação por disco:

```
sudo btrfs device usage {{caminho/para/  
sistema_de_arquivos_btrfs}}
```

# btrfs filesystem

Gerencia sistemas de arquivos btrfs.

Mais informações: <https://btrfs.readthedocs.io/en/latest/btrfs-filesystem.html>.

- Mostra uso do sistema de arquivos (opcionalmente execute como root para mostrar informações detalhadas):

```
btrfs filesystem usage {{caminho/para/montagem_btrfs}}
```

- Mostra uso por dispositivos individuais:

```
sudo btrfs filesystem show {{caminho/para/montagem_btrfs}}
```

- Desfragmenta um único arquivo em um sistema de arquivos btrfs (evite enquanto um agente de deduplicação estiver em execução):

```
sudo btrfs filesystem defragment -v {{caminho/para/arquivo}}
```

- Desfragmenta um diretório recursivamente (não cruza os limites do subvolume):

```
sudo btrfs filesystem defragment -v -r {{caminho/para/diretório}}
```

- Força a sincronização de blocos de dados não gravados com o(s) disco(s):

```
sudo btrfs filesystem sync {{caminho/para/montagem_btrfs}}
```

- Resume o uso do disco para os arquivos em um diretório recursivamente:

```
sudo btrfs filesystem du --summarize {{caminho/para/diretório}}
```

# btrfs inspect-internal

Consulta informações internas de um sistema de arquivos btrfs.

Mais informações: <https://btrfs.readthedocs.io/en/latest/btrfs-inspect-internal.html>.

- Imprime informações de superblocos:

```
sudo btrfs inspect-internal dump-super {{caminho/para/
partição}}
```

- Imprime as informações do superbloco e de todas as suas cópias:

```
sudo btrfs inspect-internal dump-super --all {{caminho/para/
partição}}
```

- Imprime informações de metadados do sistema de arquivos:

```
sudo btrfs inspect-internal dump-tree {{caminho/para/
partição}}
```

- Imprime lista de arquivos no n-ésimo inode:

```
sudo btrfs inspect-internal inode-resolve {{n}} {{caminho/
para/montagem_btrfs}}
```

- Imprime a lista de arquivos em um determinado endereço lógico:

```
sudo btrfs inspect-internal logical-resolve
{{endereço_lógico}} {{caminho/para/montagem_btrfs}}
```

- Imprime as estatísticas das árvores raiz, extensão, csum e fs:

```
sudo btrfs inspect-internal tree-stats {{caminho/para/
partição}}
```



# btrfs property

Obtém, define ou lista propriedades para um determinado objeto de sistema de arquivos btrfs (arquivos, diretórios, subvolumes, sistemas de arquivos ou dispositivos).

Mais informações: <https://btrfs.readthedocs.io/en/latest/btrfs-property.html>.

- Lista as propriedades disponíveis (e descrições) para o objeto btrfs fornecido:

```
sudo btrfs property list {{caminho/para/objeto_btrfs}}
```

- Obtém todas as propriedades para o objeto btrfs fornecido:

```
sudo btrfs property get {{caminho/para/objeto_btrfs}}
```

- Obtém a propriedade `label` (etiqueta) para o sistema de arquivos ou dispositivo btrfs fornecido:

```
sudo btrfs property get {{caminho/para/sistema_de_arquivos_btrfs}} label
```

- Obtém todas as propriedades específicas do tipo de objeto para o sistema de arquivos ou dispositivo btrfs fornecido:

```
sudo btrfs property get -t {{subvol|filesystem|inode|device}} {{caminho/para/sistema_de_arquivos_btrfs}}
```

- Define a propriedade de `compression` (compactação) para um determinado inode btrfs (um arquivo ou diretório):

```
sudo btrfs property set {{caminho/para/inode_btrfs}} compression {{zstd|zlib|lzo|none}}
```

# btrfs rescue

Tenta recuperar um sistema de arquivos btrfs danificado.

Mais informações: <https://btrfs.readthedocs.io/en/latest/btrfs-rescue.html>.

- Reconstrói a árvore de metadados do sistema de arquivos (muito lento):

```
sudo btrfs rescue chunk-recover {{caminho/para/partição}}
```

- Corrige problemas relacionados ao alinhamento do tamanho do dispositivo (por exemplo, incapaz de montar o sistema de arquivos com incompatibilidade de super total de bytes):

```
sudo btrfs rescue fix-device-size {{caminho/para/partição}}
```

- Recupera um superblock corrompido das cópias corretas (recupere a raiz da árvore do sistema de arquivos):

```
sudo btrfs rescue super-recover {{caminho/para/partição}}
```

- Recupera-se de uma transação interrompida (corrige problemas de repetição de log):

```
sudo btrfs rescue zero-log {{caminho/para/partição}}
```

- Cria um dispositivo de controle `/dev/btrfs-control` quando o `mknod` não estiver instalado:

```
sudo btrfs rescue create-control-device
```

# btrfs restore

Tenta salvar arquivos de um sistema de arquivos btrfs danificado.

Mais informações: <https://btrfs.readthedocs.io/en/latest/btrfs-restore.html>.

- Restaurar todos os arquivos de um sistema de arquivos btrfs para um determinado diretório:

```
sudo btrfs restore {{caminho/para/dispositivo_btrfs}}  
{{caminho/para/diretório_alvo}}
```

- Listar (sem escrever) os arquivos a serem restaurados de um sistema de arquivos btrfs:

```
sudo btrfs restore --dry-run {{caminho/para/  
dispositivo_btrfs}} {{caminho/para/diretório_alvo}}
```

- Restaurar arquivos correspondentes a determinados padrões regex ([c]ase-insensitive) de um sistema de arquivos btrfs (todos os diretórios pai do(s) arquivo(s) de destino também devem corresponder):

```
sudo btrfs restore --path-regex {{regex}} -c {{caminho/para/  
dispositivo_btrfs}} {{caminho/para/diretório_alvo}}
```

- Restaurar arquivos de um sistema de arquivos btrfs usando um bytenr específico da árvore raiz (consulte `btrfs - find - root`):

```
sudo btrfs restore -t {{bytenr}} {{caminho/para/  
dispositivo_btrfs}} {{caminho/para/diretório_alvo}}
```

- Restaurar arquivos de um sistema de arquivos btrfs (juntamente com metadados, atributos estendidos e Symlinks), sobrescrevendo arquivos no destino:

```
sudo btrfs restore --metadata --xattr --symlinks --overwrite  
{{caminho/para/dispositivo_btrfs}} {{caminho/para/  
diretório_alvo}}
```

# btrfs scrub

Varre os sistemas de arquivos btrfs para verificar a integridade dos dados.

Recomenda-se fazer uma varredura uma vez por mês.

Mais informações: <https://btrfs.readthedocs.io/en/latest/btrfs-scrub.html>.

- Começar uma varredura:

```
sudo btrfs scrub start {{caminho/para/  
ponto_de_montagem_btrfs}}
```

- Mostrar o status de uma varredura em andamento ou concluída:

```
sudo btrfs scrub status {{caminho/para/  
ponto_de_montagem_btrfs}}
```

- Cancelar uma varredura em andamento:

```
sudo btrfs scrub cancel {{caminho/para/  
ponto_de_montagem_btrfs}}
```

- Retomar uma varredura cancelada anteriormente:

```
sudo btrfs scrub resume {{caminho/para/  
ponto_de_montagem_btrfs}}
```

- Iniciar uma varredura, mas espera até que a varredura termine antes de sair:

```
sudo btrfs scrub start -B {{caminho/para/  
ponto_de_montagem_btrfs}}
```

- Iniciar uma varredura no modo silencioso (não imprime erros ou estatísticas):

```
sudo btrfs scrub start -q {{caminho/para/  
ponto_de_montagem_btrfs}}
```

# btrfs subvolume

Gerencia subvolumes e snapshots btrfs.

Mais informações: <https://btrfs.readthedocs.io/en/latest/btrfs-subvolume.html>.

- Cria um novo subvolume vazio:

```
sudo btrfs subvolume create {{caminho/para/novo_subvolume}}
```

- Lista todos os subvolumes e snapshots no sistema de arquivos especificado:

```
sudo btrfs subvolume list {{caminho/para/sistema_de_arquivos_btrfs}}
```

- Exclui um subvolume:

```
sudo btrfs subvolume delete {{caminho/para/subvolume}}
```

- Cria um snapshot somente leitura de um subvolume existente:

```
sudo btrfs subvolume snapshot -r {{caminho/para/subvolume_de_origem}} {{caminho/para/destino}}
```

- Cria um snapshot de leitura/gravação de um subvolume existente:

```
sudo btrfs subvolume snapshot {{caminho/para/subvolume_de_origem}} {{caminho/para/destino}}
```

- Mostra informações detalhadas sobre um subvolume:

```
sudo btrfs subvolume show {{caminho/para/subvolume}}
```

# btrfs version

Exibe a versão do btrfs-progs.

Mais informações: <https://btrfs.readthedocs.io/en/latest/btrfs.html>.

- Exibir a versão do btrfs-progs:

```
btrfs version
```

- Exibir a ajuda:

```
btrfs version --help
```

# btrfs

Um sistema de arquivos baseado no princípio copy-on-write (COW) para Linux.

Alguns subcomandos como **btrfs device** têm sua própria documentação de uso.

Mais informações: <https://btrfs.readthedocs.io/en/latest/btrfs.html>.

- Cria subvolume:

```
sudo btrfs subvolume create {{caminho/para/subvolume}}
```

- Lista subvolumes:

```
sudo btrfs subvolume list {{caminho/para/ponto_de_montagem}}
```

- Mostra informações de uso do espaço:

```
sudo btrfs filesystem df {{caminho/para/ponto_de_montagem}}
```

- Ativa a cota:

```
sudo btrfs quota enable {{caminho/para/subvolume}}
```

- Mostra a cota:

```
sudo btrfs qgroup show {{caminho/para/subvolume}}
```

# cal

Exibe as informações do calendário, destacando o dia atual.

Mais informações: <https://manned.org/cal>.

- Exibir o calendário do mês atual:

```
cal
```

- Exibir o calendário do meses anterior, atual e seguinte:

```
cal -3
```

- Utilizar segunda-feira como o primeiro dia da semana:

```
cal --monday
```

- Exibir o calendário de um ano específico (4 dígitos):

```
cal {{ano}}
```

- Exibir o calendário para um mês e ano específico:

```
cal {{mes}} {{ano}}
```



# calcurse

Um calendário e agenda baseados em texto para a linha de comando.

Mais informações: <https://calcurse.org>.

- Iniciar o calcurse em modo interativo:

```
calcurse
```

- Mostrar os agendamentos e eventos para o presente dia:

```
calcurse --appointment
```

- Apagar todos os objetos gravados localmente e importar os objetos remotos:

```
calcurse-caldav --init=keep-remote
```

- Apagar todos os objetos remotos e enviar os objetos gravados localmente:

```
calcurse-caldav --init=keep-local
```

- Copiar os objetos gravados localmente para o servidor CalDAV e vice-versa:

```
calcurse-caldav --init=two-way
```

# cat

Imprime e concatena arquivos.

Mais informações: <https://www.gnu.org/software/coreutils/cat>.

- Imprime o conteúdo de um arquivo na `stdout`:

```
cat {{caminho/para/arquivo}}
```

- Concatena vários arquivos em um arquivo de saída:

```
cat {{caminho/para/arquivo1 caminho/para/arquivo2 ...}} > {{caminho/para/arquivo_de_saida}}
```

- Anexa vários arquivos a um arquivo de saída:

```
cat {{caminho/para/arquivo1 caminho/para/arquivo2 ...}} >> {{caminho/para/arquivo_de_saida}}
```

- Copia o conteúdo de um arquivo para um arquivo de saída no modo sem buffer:

```
cat -u {{/dev/tty12}} > {{/dev/tty13}}
```

- Escreve a `stdin` em um arquivo:

```
cat - > {{caminho/para/arquivo}}
```

- [n]umera todas as linhas de saída:

```
cat -n {{caminho/para/arquivo}}
```

- Exibe caracteres não imprimíveis e espaço em branco (com o prefixo M- se não for ASCII):

```
cat -v -t -e {{caminho/para/arquivo}}
```

# CC

Este comando é um pseudônimo de **gcc**.

Mais informações: <https://gcc.gnu.org>.

- Ver documentação sobre o comando original:

```
tldr gcc
```

# certbot

O agente da Let's Encrypt para obtenção e renovação de certificados TLS automaticamente.

Sucessor do **letsencrypt**.

Mais informações: <https://certbot.eff.org/docs/using.html>.

- Obter um novo certificado via autorização webroot, porém sem instalá-lo automaticamente:

```
sudo certbot certonly --webroot --webroot-path  
{{caminho_para_webroot}} --domain {{subdominio.dominio.com}}
```

- Obter um novo certificado via autorização nginx e instalá-lo automaticamente:

```
sudo certbot --nginx --domain {{subdominio.dominio.com}}
```

- Obter um novo certificado via autorização apache e instalá-lo automaticamente:

```
sudo certbot --apache --domain {{subdominio.dominio.com}}
```

- Renovar todos os certificados que expirarão em 30 dias ou menos (não esqueça de reiniciar todos os servidores que usam os certificados):

```
sudo certbot renew
```

- Simular a obtenção de um novo certificado, porém sem salvá-lo no disco rígido:

```
sudo certbot --webroot --webroot-path  
{{caminho_para_webroot}} --domain {{subdominio.dominio.com}}  
--dry-run
```

- Obter um certificado não confiável para testes:

```
sudo certbot --webroot --webroot-path  
{{caminho_para_webroot}} --domain {{subdominio.dominio.com}}  
--test-cert
```

# cgroups

Este comando é um pseudônimo de **cgclassify**.

Mais informações: <https://www.kernel.org/doc/Documentation/cgroup-v2.txt>.

- Ver documentação sobre o comando original:

```
tldr cgclassify
```

# chage

Gerencia informações de expiração de conta e senha do usuário.

Mais informações: <https://manned.org/chage>.

- Exibir as informações referentes a senha do usuário:

```
chage --list {{nome_do_usuario}}
```

- Habilitar a expiração da senha do usuário em 10 dias:

```
sudo chage --maxdays {{10}} {{nome_do_usuario}}
```

- Desabilitar a expiração da senha do usuário:

```
sudo chage --maxdays {{-1}} {{nome_do_usuario}}
```

- Definir a data de expiração da conta do usuário:

```
sudo chage --expiredate {{YYYY-MM-DD}} {{nome_do_usuario}}
```

- Obrigar o usuário a alterar sua senha no próximo login:

```
sudo chage --lastday {{0}} {{nome_do_usuario}}
```

# chattr

Altera os atributos de arquivos ou diretórios.

Mais informações: <https://manned.org/chattr>.

- Bloquear um arquivo ou diretório para mudanças ou remoção, mesmo para um super usuário:

```
chattr +i {{caminho_do_arquivo_ou_diretorio}}
```

- Desbloquear um arquivo ou diretório:

```
chattr -i {{caminho_do_arquivo_ou_diretorio}}
```

- Bloquear diretório e todos os seus arquivos para mudanças ou remoção:

```
chattr -R +i {{caminho_do_diretorio}}
```

# chkconfig

Gerencia o runlevel dos serviços no CentOS 6.

Mais informações: <https://manned.org/chkconfig>.

- Exibir os serviços com os respectivos runlevels:

```
chkconfig --list
```

- Exibir o runlevel de um serviço:

```
chkconfig --list {{ntpd}}
```

- Habilitar o início de um serviço durante o processo de boot:

```
chkconfig {{sshd}} on
```

- Habilitar o início do serviço durante o processo de boot para os runlevels 2, 3, 4 e 5:

```
chkconfig --level {{2345}} {{sshd}} on
```

- Desabilitar a inicialização de um determinado serviço durante o processo de boot:

```
chkconfig {{ntpd}} off
```

- Desabilitar a inicialização de um determinado serviço durante o processo de boot para o runlevel 3:

```
chkconfig --level {{3}} {{ntpd}} off
```



# cmus

Player de música via terminal.

Use as setas para navegar, **<enter/return>** para selecionar, e números 1-8 para alterar as telas de opções.

Mais informações: <https://cmus.github.io>.

- Iniciar o cmus em um diretório específico:

```
cmus {{caminho_do_diretorio}}
```

- Adicionar arquivo/diretório a biblioteca:

```
:add {{caminho_para_arquivo_ou_diretorio}}
```

- Parar/reiniciar a música atual:

```
c
```

- Ativar/Desativar o modo aleatório:

```
s
```

- Sair cmus:

```
q
```

# cp

Copia arquivos e diretórios.

Mais informações: <https://www.gnu.org/software/coreutils/cp>.

- Copiar um arquivo para outra localização:

```
cp {{caminho/para/arquivo_entrada.ext}} {{caminho/para/arquivo_saída.ext}}
```

- Copiar um arquivo para dentro de outro diretório, mantendo o nome:

```
cp {{caminho/para/arquivo.ext}} {{caminho/para/diretório}}
```

- Recursivamente copiar os conteúdos de um diretório para outra localização (se a destinação existe, o diretório é copiado para dentro dela):

```
cp -r {{caminho/para/diretório_fonte}} {{caminho/para/diretório_alvo}}
```

- Copiar um diretório recursivamente, em modo verboso (mostra os arquivos conforme eles são copiados):

```
cp -vr {{caminho/para/diretório_fonte}} {{caminho/para/diretório_alvo}}
```

- Copiar arquivos de texto para outra localização, em modo interativo (exige confirmação do usuário antes de sobrescrever):

```
cp -i {{*.txt}} {{caminho/para/diretório_alvo}}
```

- Seguir links simbólicos antes de copiar:

```
cp -L {{link}} {{caminho/para/diretório_alvo}}
```

- Usar todo o caminho dos arquivos fonte, criando quaisquer diretórios intermediários ausentes quando copia:

```
cp --parents {{fonte/caminho/para/arquivo}} {{caminho/para/arquivo_alvo}}
```

# cpuid

Exibe informações detalhadas sobre todas as CPUs.

Mais informações: <http://etallen.com/cpuid.html>.

- Exibir informações de todas as CPUs:

```
cpuid
```

- Exibir informações apenas da CPU atual:

```
cpuid -1
```

- Exibir informações em hexadecimal sem decodificação:

```
cpuid -r
```

# csplit

Divide um arquivo em várias partes.

O padrão de nomenclatura dos arquivos será "xx00", "xx01" e assim por diante.

Mais informações: <https://www.gnu.org/software/coreutils/csplit>.

- Dividir um arquivo nas linhas 5 e 23:

```
csplit {{arquivo}} {{5}} {{23}}
```

- Dividir um arquivo a cada 5 linhas (este comando irá falhar se o total de linhas do arquivo não for divisível por 5):

```
csplit {{arquivo}} {{5}} {*}
```

- Dividir um arquivo a cada 5 linhas, ignorando o fato do total de linhas ser divisível por 5:

```
csplit -k {{arquivo}} {{5}} {*}
```

- Dividir o arquivo na linha 5 e utilizar um prefixo específico para os arquivos de saída:

```
csplit {{arquivo}} {{5}} -f {{prefix}}
```

- Dividir um arquivo na linha que atenda a expressão regular:

```
csplit {{arquivo}} /{{expressao_regular}}/
```

# daemon

Roda processos em daemons.

Mais informações: <https://manned.org/man/daemon.1>.

- Roda um comando como um daemon:

```
daemon --name="{{nome}}" {{comando}}
```

- Roda um comando como um daemon que será reiniciado se o comando falhar:

```
daemon --name="{{nome}}" --respawn {{comando}}
```

- Roda um comando como um daemon que será reiniciado se falar, com duas tentativas a cada 10 segundos:

```
daemon --name="{{nome}}" --respawn --attempts=2 --delay=10  
{{comando}}
```

- Roda um comando como um daemon, gravando registros em um arquivo específico:

```
daemon --name="{{nome}}" --errlog={{caminho/para/  
arquivo.log}} {{comando}}
```

- Elimina um daemon (SIGTERM):

```
daemon --name="{{nome}}" --stop
```

- Lista os daemons:

```
daemon --list
```

# dconf read

Lê valores de chave dos bancos de dados dconf.

Veja também: **dconf**.

Mais informações: <https://manned.org/dconf>.

- Imprime um valor de chave específico:

```
dconf read {{/caminho/para/chave}}
```

- Imprime o valor padrão de uma chave específica:

```
dconf read -d {{/caminho/para/chave}}
```

# dconf reset

Redefine chaves valores nos bancos de dados dconf.

Veja também: **dconf**.

Mais informações: <https://manned.org/dconf>.

- Redefine um valor de chave específico:

```
dconf read {{/caminho/para/chave}}
```

- Redefine um diretório específico:

```
dconf read -d {{/caminho/para/diretório/}}
```

# dconf write

Escreve valores de chave nos bancos de dados dconf.

Veja também: **dconf**.

Mais informações: <https://manned.org/dconf>.

- Escreve um valor de chave específico:

```
dconf write {{/caminho/para/chave}} "{{valor}}"
```

- Escreve uma string específica como valor de chave:

```
dconf write {{/caminho/para/chave}} "'{{string}}'"
```

- Escreve um inteiro específico como valor de chave:

```
dconf write {{/caminho/para/chave}} "{{5}}"
```

- Escreve um booleano específico como valor de chave:

```
dconf write {{/caminho/para/chave}} "{{true|false}}"
```

- Escreve um array específico como valor de chave:

```
dconf write {{/caminho/para/chave}} "[{{'primeiro',  
'segundo', ...}}]"
```

- Escreve um array vazio específico como valor de chave:

```
dconf write {{/caminho/para/chave}} "@as []"
```



# dconf

Gerencia banco de dados dconf.

Veja também: **dconf-read**, **dconf-reset**, **dconf-write**, **gsettings**.

Mais informações: <https://manned.org/dconf>.

- Imprime um valor de chave específico:

```
dconf read {{/caminho/para/chave}}
```

- Imprime sub-diretórios e sub-chaves de um caminho específico:

```
dconf list {{/caminho/para/diretório/}}
```

- Grava um valor de chave específico:

```
dconf write {{/caminho/para/chave}} "{{{valor}}}"
```

- Redefine um valor de chave específico:

```
dconf reset {{/caminho/para/chave}}
```

- Observa alterações em uma chave/diretório específico:

```
dconf watch {{/caminho/para/chave|/caminho/para/diretório/}}
```

- Despeja um diretório específico no formato de arquivo INI:

```
dconf dump {{/caminho/para/diretório/}}
```

# diff3

Compara 3 arquivos linha por linha.

Mais informações: [https://www.gnu.org/software/diffutils/manual/html\\_node/Invoking-diff3.html](https://www.gnu.org/software/diffutils/manual/html_node/Invoking-diff3.html).

- Comparar os arquivos:

```
diff3 {{caminho/para/arquivo1}} {{caminho/para/arquivo2}}  
{{caminho/para/arquivo3}}
```

- Exibir todas as diferenças, destacando os conflitos:

```
diff3 --show-all {{caminho/para/arquivo1}} {{caminho/para/  
arquivo2}} {{caminho/para/arquivo3}}
```

# distrobox-create

Criar um contêiner distrobox. Veja também: **tldr distrobox**.

O contêiner criado será integrado ao sistema host, permitindo o compartilhamento do diretório HOME do usuário, armazenamento externo, dispositivos USB externos, aplicativos gráficos (X11/Wayland) e áudio.

Mais informações: <https://distrobox.it/usage/distrobox-create>.

- Criar um contêiner distrobox usando a imagem do Ubuntu:

```
distrobox-create {{nome_do_contêiner}} --image  
{{ubuntu:latest}}
```

- Clonar um contêiner distrobox:

```
distrobox-create --clone {{nome_do_contêiner}}  
{{nome_do_contêiner_clonado}}
```

# distrobox-enter

Entrar em um contêiner distrobox. Veja também: **tldr distrobox**.

O comando padrão executado é o seu SHELL, mas você pode especificar shells diferentes ou comandos completos para serem executados.

Se usado dentro de um script, um aplicativo ou um serviço, você pode usar o modo **--headless** para desabilitar o tty e a interatividade.

Mais informações: <https://distrobox.it/usage/distrobox-enter>.

- Entrar em um contêiner distrobox:

```
distrobox-enter {{nome_do_contêiner}}
```

- Entrar em um contêiner distrobox e executar um comando no login:

```
distrobox-enter {{nome_do_contêiner}} -- {{sh -l}}
```

- Entrar em um contêiner distrobox sem instanciar um tty:

```
distrobox-enter --name {{nome_do_contêiner}} -- {{uptime -p}}
```

# distrobox-export

Exportar um aplicativo/serviço/binário do contêiner para o sistema operacional host.

Subcomando de **distrobox**. Veja também: **tldr distrobox**.

Mais informações: <https://distrobox.it/usage/distrobox-export>.

- Exportar um aplicativo do contêiner para o host (a entrada e o ícone do aplicativo aparecerão na lista de aplicativos do seu sistema host):

```
distrobox-export --app {{nome_do_pacote}} --extra-flags "--foreground"
```

- Exportar um binário do contêiner para o host:

```
distrobox-export --bin {{caminho/para/binário}} --export-path {{caminho/para/binário_no_host}}
```

- Exportar um binário do contêiner para o host (por exemplo, \$HOME/.local/bin):

```
distrobox-export --bin {{caminho/para/binário}} --export-path {{caminho/de/exportação}}
```

- Exportar um serviço do contêiner para o host (--sudo executará o serviço como root dentro do contêiner):

```
distrobox-export --service {{pacote}} --extra-flags "--allow-newer-config" --sudo
```

- Desexportar/deletar um aplicativo exportado:

```
distrobox-export --app {{pacote}} --delete
```

# distrobox-host-exec

Executar um comando no sistema host a partir do interior de um contêiner distrobox.

Subcomando de **distrobox**. Veja também: **tldr distrobox**.

Mais informações: <https://distrobox.it/usage/distrobox-host-exec>.

- Executar um comando no sistema host a partir do interior do contêiner distrobox:

```
distrobox-host-exec "{{comando}}"
```

- Executar o comando `ls` no sistema host a partir do interior do contêiner:

```
distrobox-host-exec ls
```

# distrobox-list

Listar todos os contêineres distrobox. Veja também: **tldr distrobox**.

Os contêineres distrobox são listados separadamente dos demais contêineres normais do Podman ou Docker.

Mais informações: <https://distrobox.it/usage/distrobox-list>.

- Listar todos os contêineres distrobox:

```
distrobox-list
```

- Listar todos os contêineres distrobox com informações detalhadas:

```
distrobox-list --verbose
```

# distrobox-rm

Remover um contêiner distrobox.

Subcomando de **distrobox**. Veja também: **tldr distrobox**.

Mais informações: <https://distrobox.it/usage/distrobox-rm>.

- Remover um contêiner distrobox (Dica: Pare o contêiner antes de removê-lo):

```
distrobox-rm {{nome_do_contêiner}}
```

- Remover um contêiner distrobox forçadamente:

```
distrobox-rm {{nome_do_contêiner}} --force
```



# distrobox-stop

Parar um contêiner distrobox.

Subcomando de **distrobox**. Veja também: **tldr distrobox**.

Mais informações: <https://distrobox.it/usage/distrobox-stop>.

- Parar um contêiner distrobox:

```
distrobox-stop {{nome_do_contêiner}}
```

- Parar um contêiner distrobox de forma não interativa (sem confirmação):

```
distrobox-stop --name {{nome_do_contêiner}} --yes
```

# distrobox-upgrade

Atualizar um ou vários contêineres distrobox.

Subcomando de **distrobox**. Veja também: **tldr distrobox**.

Mais informações: <https://distrobox.it/usage/distrobox-upgrade>.

- Atualizar um contêiner usando o gerenciador de pacotes nativo do contêiner:

```
distrobox-upgrade {{nome_do_contêiner}}
```

- Atualizar todos os contêineres usando os gerenciadores de pacotes nativos dos contêineres:

```
distrobox-upgrade --all
```

- Atualizar contêineres específicos via o gerenciador de pacotes nativo do contêiner:

```
distrobox-upgrade {{contêiner1 contêiner2 ...}}
```

# distrobox

Use qualquer distribuição Linux dentro do seu terminal em um contêiner. Instale e use pacotes dentro dele, integrando-se perfeitamente ao sistema operacional host, compartilhando armazenamento (diretório "home") e hardware.

Ele utiliza o Podman ou Docker para criar seus contêineres.

Mais informações: <https://github.com/89luca89/distrobox>.

- Visualizar a documentação para criar contêineres:

```
tldr distrobox-create
```

- Visualizar a documentação para listar informações do contêiner:

```
tldr distrobox-list
```

- Visualizar a documentação para entrar no contêiner:

```
tldr distrobox-enter
```

- Visualizar a documentação para executar um comando no host a partir do interior de um contêiner:

```
tldr distrobox-host-exec
```

- Visualizar a documentação para exportar aplicativos/serviços/binários do contêiner para o host:

```
tldr distrobox-export
```

- Visualizar a documentação para atualizar contêineres:

```
tldr distrobox-upgrade
```

- Visualizar a documentação para parar os contêineres:

```
tldr distrobox-stop
```

- Visualizar a documentação para remover os contêineres:

```
tldr distrobox-rm
```

# dmesg

Escreve as mensagens do kernel na terminal.

Mais informações: <https://manned.org/dmesg>.

- Exibir as mensagens do kernel:

```
dmesg
```

- Exibir as mensagens de erro do kernel:

```
dmesg --level err
```

- Exibir as mensagens do kernel e manter o terminal esperando por novas mensagens, semelhante ao `tail -f` (disponível nas versões 3.5.0 e superiores do kernel):

```
dmesg -w
```

- Exibir a quantidade de memória física disponível no sistema:

```
dmesg | grep -i memory
```

- Exibir as mensagens do kernel divididas em páginas:

```
dmesg | less
```

- Exibir as mensagens do kernel com data/hora (disponível nas versões 3.5.0 e superiores do kernel):

```
dmesg -T
```

- Exibir as mensagens do kernel em um formato de fácil leitura (disponível nas versões 3.5.0 e superiores do kernel):

```
dmesg -H
```

- Exibir as mensagens do kernel utilizando cores (disponível nas versões 3.5.0 e superiores do kernel):

```
dmesg -L
```

# dmidecode

Exibe em formato de fácil leitura o sumário DMI (também conhecido como SMBIOS) .

Requer privilégio de super usuário.

Mais informações: <https://manned.org/dmidecode>.

- Exibir o sumário do DMI:

```
sudo dmidecode
```

- Exibir a versão da BIOS:

```
sudo dmidecode -s bios-version
```

- Exibir o número de série do sistema:

```
sudo dmidecode -s system-serial-number
```

- Exibir as informações da BIOS:

```
sudo dmidecode -t bios
```

- Exibir as informações da CPU:

```
sudo dmidecode -t processor
```

- Exibir as informações da memória:

```
sudo dmidecode -t memory
```

# dnf

Gerenciador de pacotes das distribuições baseadas em RHEL (substituto do yum).

Mais informações: <https://dnf.readthedocs.io>.

- Instalar um novo pacote:

```
sudo dnf install {{nome_do_pacote}}
```

- Instalar um novo pacote e responder sim para todas as questões:

```
sudo dnf -y install {{nome_do_pacote}}
```

- Remover um pacote:

```
sudo dnf remove {{nome_do_pacote}}
```

- Atualizar todos os pacotes instalados para as versões mais recentes:

```
sudo dnf upgrade
```

# dockerd

Um processo persistente para iniciar e gerenciar contêineres Docker.

Mais informações: <https://docs.docker.com/engine/reference/commandline/dockerd/>.

- Executar o daemon do Docker:

```
dockerd
```

- Executar o daemon do Docker e configurá-lo para escutar em sockets específicos (UNIX e TCP):

```
dockerd --host unix://{{caminho/para/tmp.sock}} --host tcp://{{ip}}
```

- Executar com um arquivo PID específico para o daemon:

```
dockerd --pidfile {{caminho/para/arquivo_pid}}
```

- Executar no modo de depuração:

```
dockerd --debug
```

- Executar e definir um nível de log específico:

```
dockerd --log-level={{debug|info|warn|error|fatal}}
```

# dolphin

Gerenciador de arquivos do KDE.

Mais informações: <https://apps.kde.org/dolphin/>.

- Inicializa o Dolphin:

```
dolphin
```

- Inicializa o Dolphin em um diretório específico:

```
dolphin {{caminho/para/diretorio}}
```

- Inicializa o Dolphin com um arquivo ou diretório específico:

```
dolphin --select {{caminho/para/arquivo_ou_diretorio}}
```

- Inicializa o Dolphin em uma janela separada:

```
dolphin --new-window
```

- Inicializa o Dolphin no modo de divisão de tela:

```
dolphin --split
```

- Inicializa o daemon do Dolphin (necessário apenas para usar a interface do DBus):

```
dolphin --daemon
```

- Exibe log de ajuda (em inglês):

```
dolphin --help
```



# dpkg-query

Ferramenta que mostra informações dos pacotes instalados.

Mais informações: <https://manpages.debian.org/latest/dpkg/dpkg-query.1.html>.

- Exibir os pacotes instalados:

```
dpkg-query -l
```

- Exibir os pacotes instalados correspondentes ao critério de busca:

```
dpkg-query -l '{{criterio_de_busca}}'
```

- Exibir todos os arquivos instalados por um pacote:

```
dpkg-query -L {{nome_do_pacote}}
```

- Exibir informações sobre um pacote:

```
dpkg-query -s {{nome_do_pacote}}
```

# dpkg

Gerenciador de pacotes Debian.

Alguns subcomandos como **dpkg deb** tem sua própria documentação de uso.

Mais informações: <https://manpages.debian.org/latest/dpkg/dpkg.html>.

- Instalar um pacote:

```
dpkg -i {{arquivo.deb}}
```

- Remover um pacote:

```
dpkg -r {{nome_do_pacote}}
```

- Exibir os pacotes correspondentes ao critério de busca:

```
dpkg -l {{critério_de_busca}}
```

- Exibe o conteúdo do pacote:

```
dpkg -L {{nome_do_pacote}}
```

- Exibir o conteúdo do arquivo de um pacote:

```
dpkg -c {{arquivo.deb}}
```

- Apresentar o pacote proprietário de um determinado arquivo:

```
dpkg -S {{nome_do_arquivo}}
```

# eyeD3

Lê e manipula os metadados de arquivos MP3.

Mais informações: <https://eyed3.readthedocs.io>.

- Visualizar as informações de um arquivo MP3:

```
eyeD3 {{arquivo.mp3}}
```

- Definir o título de um arquivo MP3:

```
eyeD3 --title "{{título}}" {{arquivo.mp3}}
```

- Definir o álbum de todos os arquivos MP3 de um diretório:

```
eyeD3 --album "{{nome_do_album}}" {{*.mp3}}
```

- Definir a capa do álbum para um arquivo MP3:

```
eyeD3 --add-image {{capa.jpeg}}:FRONT_COVER: {{arquivo.mp3}}
```

# fatlabel

Define ou exibe o rótulo de uma partição FAT32.

Mais informações: <https://manned.org/fatlabel>.

- Exibir o rótulo de uma partição FAT32:

```
fatlabel {{/dev/sda1}}
```

- Definir o rótulo de uma partição FAT32:

```
fatlabel {{/dev/sdc3}} "{{rotulo}}"
```

# fdisk

Gerenciador de tabelas de partições e partições no disco rígido.

Mais informações: <https://manned.org/fdisk>.

- Exibir as partições:

```
fdisk -l
```

- Iniciar o manipulador de partições:

```
fdisk {{/dev/sda}}
```

# flameshot

Função de captura da tela com uma Interface Gráfica do Usuário.

Suporta edição básica de imagens, como texto, formas, cores e imgur.

Mais informações: <https://flameshot.org>.

- Cria uma captura da tela completa:

```
flameshot full
```

- Cria uma captura da tela interativamente:

```
flameshot gui
```

- Cria uma captura da tela e salva em um caminho específico:

```
flameshot gui --path {{caminho/para/diretório}}
```

- Cria uma captura da tela interativamente em um modo simplificado:

```
flameshot launcher
```

- Cria uma captura da tela a partir de um monitor específico:

```
flameshot screen --number {{2}}
```

- Cria uma captura da tela e imprime na saída padrão:

```
flameshot gui --raw
```

- Cria uma captura da tela e copia para a área de transferência:

```
flameshot gui --clipboard
```

- Cria uma captura da tela com um atraso específico em milissegundos:

```
flameshot full --delay {{5000}}
```

# flatpak

Construa, instale e execute aplicações e plataformas flatpak.

Mais informações: <https://docs.flatpak.org/en/latest/flatpak-command-reference.html#flatpak>.

- Executar uma aplicação instalada:

```
flatpak run {{nome}}
```

- Instalar uma aplicação de uma fonte remota:

```
flatpak install {{remoto}} {{nome}}
```

- Listar todas as aplicações e plataformas instaladas:

```
flatpak list
```

- Atualizar todas as aplicações e plataformas instaladas:

```
flatpak update
```

- Adicionar uma fonte remota:

```
flatpak remote-add --if-not-exists {{nome_remoto}}  
{{url_remoto}}
```

- Listar todas fontes remotas configuradas:

```
flatpak remote-list
```

- Remover uma aplicação instalada:

```
flatpak remove {{nome}}
```

- Mostrar informações sobre uma aplicação instalada:

```
flatpak info {{nome}}
```

# free

Mostra a quantidade de memória livre e utilizada pelo sistema.

Mais informações: <https://manned.org/free>.

- Mostrar memória do sistema:

```
free
```

- Mostrar memória em Bytes/KB/MB/GB:

```
free -{{b|k|m|g}}
```

- Mostrar memória em unidades legíveis por seres humanos:

```
free -h
```

- Atualizar a saída padrão a cada 2 segundos:

```
free -s {{2}}
```



# genisoimage

Programa de pré-masterização para gerar sistemas de arquivos híbridos ISO9660/Joliet/HFS.

Mais informações: <https://manpages.debian.org/latest/genisoimage/genisoimage.1.en.html>.

- Criar uma imagem ISO a partir do diretório de origem fornecido:

```
genisoimage -o {{minhaimagem.iso}} {{caminho/para/diretório_origem}}
```

- Criar uma imagem ISO com arquivos maiores que 2GiB, relatando um tamanho aparente menor para o sistema de arquivos ISO9660:

```
genisoimage -o -allow-limited-size {{minhaimagem.iso}}  
{{caminho/para/diretório_origem}}
```

# gnome-calculator

A calculadora oficial para o ambiente de desktop GNOME.

Mais informações: <https://wiki.gnome.org/Apps/Calculator>.

- Inicia a GNOME Calculator GUI:

```
gnome-calculator
```

- Resolve uma equação específica na linha de comando sem iniciar o aplicativo de desktop:

```
gnome-calculator --solve {{2^5 * 2 + 5}}
```

- Exibe a versão:

```
gnome-calculator --version
```

# gnome-terminal

O emulador do Terminal GNOME.

Mais informações: <https://help.gnome.org/users/gnome-terminal/stable/>.

- Abre uma nova janela de terminal GNOME:

```
gnome-terminal
```

- Executa um comando específico em uma nova janela de terminal:

```
gnome-terminal -- {{commando}}
```

- Abre uma nova aba na última janela aberta:

```
gnome-terminal --tab
```

- Define o título da nova aba:

```
gnome-terminal --tab --title "{{título}}"
```

# gsettings

Consulta e modifica configurações do dconf com validação de esquema.

Mais informações: [https://access.redhat.com/documentation/en-us/red\\_hat\\_enterprise\\_linux/8/html/using\\_the\\_desktop\\_environment\\_in\\_rhel\\_8/configuring-gnome-at-low-level\\_using-the-desktop-environment-in-rhel-8#using-gsettings-command\\_configuring-gnome-at-low-level](https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html/using_the_desktop_environment_in_rhel_8/configuring-gnome-at-low-level_using-the-desktop-environment-in-rhel-8#using-gsettings-command_configuring-gnome-at-low-level).

- Define o valor de uma chave. Falha se a chave não existe ou o valor está fora do intervalo:

```
gsettings set {{org.exemplo.esquema}} {{chave-exemplo}}  
{{valor}}
```

- Imprime o valor de uma chave ou o padrão fornecido pelo esquema se a chave não foi definida no dconf:

```
gsettings get {{org.exemplo.esquema}} {{chave-exemplo}}
```

- Desfaz a definição de uma chave, para que o valor padrão do esquema seja usado:

```
gsettings reset {{org.exemplo.esquema}} {{chave-exemplo}}
```

- Exibe todos os esquemas, chaves e valores (não realocáveis):

```
gsettings list-recursively
```

- Exibe todas as chaves e valores (padrão se não definido) de um esquema:

```
gsettings list-recursively {{org.exemplo.esquema}}
```

- Exibe valores permitidos pelo esquema para uma chave (útil com chaves enumeráveis):

```
gsettings range {{org.exemplo.esquema}} {{chave-exemplo}}
```

- Exibe a descrição legível por humanos de uma chave:

```
gsettings describe {{org.exemplo.esquema}} {{chave-exemplo}}
```

# i3lock

Bloqueador de tela simples para o i3wm.

Mais informações: <https://i3wm.org/i3lock>.

- Bloqueia a tela com uma cor de fundo (formato rrggbb):

```
i3lock -c {{0000ff}}
```

- Bloqueia a tela com uma imagem PNG:

```
i3lock -i {{local/da/imagem.png}}
```

- Desabilita o indicador de desbloqueio (remove a resposta ao apertar teclas):

```
i3lock -u
```

- Exibe o ponteiro do mouse ao invés de ocultá-lo ('default' para o ponteiro padrão, 'win' para um ponteiro MS Windows):

```
i3lock -p {{default|win}}
```

- Bloqueia a tela com uma imagem PNG exibida em múltiplos monitores, com o ponteiro do mouse habilitado:

```
i3lock -i {{local/da/imagem.png}} -p {{default|win}} -t
```

# ip-route-list

Este comando é um pseudônimo de **ip-route-show**.

- Ver documentação sobre o comando original:

```
tldr ip-route-show
```

# ip

Mostra / manipula roteamento, dispositivos, roteamento baseado em políticas e túneis.

Alguns subcomandos como **ip address** têm suas próprias documentações de uso.

Mais informações: <https://www.man7.org/linux/man-pages/man8/ip.8.html>.

- Lista interfaces com informações detalhadas:

```
ip address
```

- Lista interfaces com breves informações sobre a camada de rede:

```
ip -brief address
```

- Lista interfaces com breves informações sobre a camada de link de dados:

```
ip -brief link
```

- Exibe a tabela de roteamento:

```
ip route
```

- Mostra vizinhos (ARP tabela):

```
ip neighbour
```

- Ativa / desativa uma interface:

```
ip link set {{interface}} up/down
```

- Adiciona / remove um endereço de IP a uma interface:

```
ip addr add/del {{ip}}/{{mask}} dev {{interface}}
```

- Adiciona uma rota padrão:

```
ip route add default via {{ip}} dev {{interface}}
```

# kdocker

Ancorar facilmente aplicativos à bandeja do sistema.

Mais informações: <https://github.com/user-none/KDocker>.

- Exibir um cursor para enviar uma janela para a bandeja do sistema ao pressionar o botão esquerdo do mouse (pressione qualquer outro botão do mouse para cancelar):

```
kdocker
```

- Abrir um aplicativo e enviá-lo para a bandeja do sistema:

```
kdocker {{aplicativo}}
```

- Enviar a janela com foco para a bandeja do sistema:

```
kdocker -f
```

- Exibir um cursor para enviar uma janela para a bandeja do sistema com um ícone personalizado ao pressionar o botão esquerdo do mouse:

```
kdocker -i {/caminho/para/ícone}
```

- Abrir um aplicativo, enviá-lo para a bandeja do sistema e, se perder o foco, minimizá-lo:

```
kdocker -l {{aplicativo}}
```

- Exibir a versão:

```
kdocker --version
```



# locale

Obtém informações específicas da localidade.

Mais informações: <https://manned.org/locale>.

- Lista todas as variáveis globais de ambiente que descrevem a localidade do usuário:

```
locale
```

- Lista todas as localidades disponíveis:

```
locale --all-locales
```

- Exibe todas as localidades disponíveis e os metadados associados:

```
locale --all-locales --verbose
```

- Exibe o formato da data atual:

```
locale date_fmt
```

# locate

Encontre nomes de arquivos rapidamente.

Mais informações: <https://manned.org/locate>.

- Procura por padrões no banco de dados. Nota: o banco de dados é recalculado periodicamente (geralmente semanalmente ou diariamente):

```
locate {{padrão}}
```

- Procura um arquivo pelo seu nome de arquivo exato(um padrão que não contém caracteres curingas é interpretado como `*pattern*`):

```
locate */{{nome_do_arquivo}}
```

- Recalcula o banco de dados. Você precisa fazer se você quiser achar os arquivos recentemente adicionados:

```
sudo updatedb
```

# lsblk

Lista informações sobre dispositivos.

Mais informações: <https://manned.org/lsblk>.

- Lista todos dispositivos de armazenamento no formato de árvore:

```
lsblk
```

- Também lista dispositivos vazios:

```
lsblk -a
```

- Mostrar a coluna de tamanhos em bytes, em vez de um formato legível por humanos:

```
lsblk -b
```

- Mostrar na saída padrão informações sobre os filesystems dos dispositivos:

```
lsblk -f
```

- Utiliza caracteres ASCII para o formato de árvore:

```
lsblk -i
```

- Mostrar na saída padrão informações sobre block-device topology:

```
lsblk -t
```

- Excluir da saída padrão os dispositivos especificados por seus respectivos números separados por vírgulas:

```
lsblk -e {{1,7}}
```

- Mostrar um resumo de forma customizada passando as colunas separadas por vírgulas:

```
lsblk --output {{NAME}},{{SERIAL}},{{MODEL}},{{TRAN}},  
{{TYPE}},{{SIZE}},{{FSTYPE}},{{MOUNTPOINT}}
```

# lscpu

Exibe informações sobre a arquitetura da CPU.

Mais informações: <https://manned.org/lscpu>.

- Exibe informações sobre todas as CPUs:

```
lscpu
```

- Exibe informações em uma tabela:

```
lscpu --extended
```

- Exibe apenas informações sobre CPUs desligadas em uma tabela:

```
lscpu --extended --offline
```

# man

Formata e exibe páginas do manual.

Mais informações: <https://manned.org/man>.

- Exibe a página do manual para um comando:

```
man {{comando}}
```

- Exibe a página do manual para um comando da seção 7:

```
man {{7}} {{comando}}
```

- Lista todas as seções disponíveis para um comando:

```
man --what-is {{comando}}
```

- Exibe o caminho pesquisado para páginas do manual:

```
man --path
```

- Exibe a localização de uma página do manual em vez da página em si:

```
man --where {{comando}}
```

- Exibe a página do manual usando uma localização específica:

```
man --locale={{localização}} {{comando}}
```

- Procura por páginas do manual que contenham uma certa string:

```
man --apropos "{{string_buscada}}"
```

# megadl

Este comando é um pseudônimo de **megatools-dl**.

Mais informações: <https://megatools.megous.com/man/megatools-dl.html>.

- Ver documentação sobre o comando original:

```
tldr megatools-dl
```

# mkfs.btrfs

Crie um sistema de arquivos btrfs.

O padrão é **raid1**, que especifica 2 cópias de um determinado bloco de dados espalhados por 2 dispositivos diferentes.

Mais informações: <https://btrfs.readthedocs.io/en/latest/mkfs.btrfs.html>.

- Criar um sistema de arquivos btrfs em um único dispositivo:

```
sudo mkfs.btrfs --metadata single --data single {{/dev/sda}}
```

- Criar um sistema de arquivos btrfs em vários dispositivos com raid1:

```
sudo mkfs.btrfs --metadata raid1 --data raid1 {{/dev/sda}}  
{{/dev/sdb}} {{/dev/sdN}}
```

- Definir um rótulo para o sistema de arquivos:

```
sudo mkfs.btrfs --label "{{rótulo}}" {{/dev/sda}} [{{/dev/  
sdN}}]
```

# mkfs.ext4

Cria um sistema de arquivos ext4 dentro de uma partição.

Mais informações: <https://manned.org/mkfs.ext4>.

- Cria um sistema de arquivos ext4 dentro da partição 1 no dispositivo b (sdb1):

```
sudo mkfs.ext4 {{/dev/sdb1}}
```

- Cria um sistema de arquivo ext4 com um rótulo de volume:

```
sudo mkfs.ext4 -L {{rótulo_de_volume}} {{/dev/sdb1}}
```



# mkfs.fat

Cria um sistema de arquivos MS-DOS dentro de uma partição.

Mais informações: <https://manned.org/mkfs.fat>.

- Cria um sistema de arquivos fat dentro da partição 1 do dispositivo b (sdb1):

```
mkfs.fat {{/dev/sdb1}}
```

- Cria um sistema de arquivos com um nome de volume:

```
mkfs.fat -n {{nome_de_volume}} {{/dev/sdb1}}
```

- Cria um sistema de arquivos com um id de volume:

```
mkfs.fat -i {{id_de_volume}} {{/dev/sdb1}}
```

- Usa 5 em vez de 2 tabelas de alocação de arquivos:

```
mkfs.fat -f 5 {{/dev/sdb1}}
```

# mkfs

Cria um sistema de arquivos Linux em uma partição do disco rígido.

Esse comando está obsoleto em favor dos utilitários mkfs. específicos de sistema de arquivos.

Mais informações: <https://manned.org/mkfs>.

- Cria um sistema de arquivo ext2 do Linux em uma partição:

```
mkfs {{caminho/para/partição}}
```

- Cria um sistema de arquivos de um tipo especificado:

```
mkfs -t {{ext4}} {{caminho/para/partição}}
```

- Cria um sistema de arquivos de um tipo especificado e verifica por blocos ruins:

```
mkfs -c -t {{ntfs}} {{caminho/para/partição}}
```

# ncal

Este comando é um pseudônimo de **cal**.

Mais informações: <https://manned.org/ncal>.

- Ver documentação sobre o comando original:

```
tldr cal
```

# pacman-mirrors

Gera uma lista de mirrors do pacman para o Manjaro Linux.

Toda execução do pacman-mirrors requer que você sincronize seu bando de dados e atualize seu sistema usado **sudo pacman -Syyu**.

Veja também: **pacman**.

Mais informações: <https://wiki.manjaro.org/index.php?title=Pacman-mirrors>.

- Gera uma lista de mirrors usando as configurações padrão:

```
sudo pacman-mirrors --fasttrack
```

- Obtém o status dos mirrors atuais:

```
pacman-mirrors --status
```

- Exibe a branch atual:

```
pacman-mirrors --get-branch
```

- Muda para uma branch diferente:

```
sudo pacman-mirrors --api --set-branch {{stable|unstable|testing}}
```

- Gera uma lista de mirror, usando apenas mirrors em seu país:

```
sudo pacman-mirrors --geoip
```

# pacman

Utilitário de Arch Linux para gerenciamento de pacotes.

Veja também: **pacman-database**, **pacman-deptest**, **pacman-files**, **pacman-key**, **pacman-mirrors**, **pacman-query**, **pacman-remove**, **pacman-sync**, **pacman-upgrade**.

Mais informações: <https://man.archlinux.org/man/pacman.8>.

- Sincroniza e atualiza todos os pacotes:

```
sudo pacman -Syu
```

- Instala um novo pacote:

```
sudo pacman -S {{nome_do_pacote}}
```

- Remove um pacote e suas dependências:

```
sudo pacman -Rs {{nome_do_pacote}}
```

- Procura no banco de dados de pacotes por uma expressão regular ou palavra-chave:

```
pacman -Ss "{{padrao_buscado}}"
```

- Lista pacotes instalados e versões:

```
pacman -Q
```

- Lista apenas os pacotes explicitamente instalados e versões:

```
pacman -Qe
```

- Lista pacotes órfãos (instalado como dependência mas não requerido por qualquer pacote):

```
pacman -Qtdq
```

- Esvazia completamente o cache do pacman:

```
sudo pacman -Scc
```

# paru

Um auxiliar do AUR e um wrapper do pacman.

Mais informações: <https://github.com/Morganamilo/paru>.

- Pesquisa e instala interativamente um pacote:

```
paru {{nome_do_pacote_ou_termo_de_pesquisa}}
```

- Sincroniza e atualizar todos os pacotes:

```
paru
```

- Atualiza pacotes do AUR:

```
paru -Sua
```

- Obtém informações sobre um pacote:

```
paru -Si {{nome_do_pacote}}
```

- Faz o download do PKGBUILD e outros arquivos de origem do pacote do AUR ou ABS:

```
paru --getpkgbuild {{nome_do_pacote}}
```

- Exibe o arquivo PKGBUILD de um pacote:

```
paru --getpkgbuild --print {{nome_do_pacote}}
```

# pi

Calcula a constante decimal de Arquimedes Pi na linha de comando.

Mais informações: <https://manned.org/pi>.

- Exibe 100 dígitos decimais da constante de Archimedes Pi:

```
pi
```

- Exibe um número específico de dígitos decimais da constante de Archimedes Pi:

```
pi {{number}}
```

- Exibe ajuda:

```
pi --help
```

- Exibe a versão:

```
pi --version
```

- Exibe leituras recomendadas:

```
pi --bibliography
```

# poweroff

Desliga o sistema.

Mais informações: <https://www.man7.org/linux/man-pages/man8/poweroff.8.html>.

- Desliga o sistema:

```
poweroff
```

- Para o sistema (mesmo que `halt`):

```
poweroff --halt
```

- Reinicia o sistema (mesmo que `reboot`):

```
poweroff --reboot
```

- Desliga imediatamente sem contato com o gerenciador do sistema:

```
poweroff --force --force
```

- Grava a entrada de desligamento wtmp sem desligar o sistema:

```
poweroff --wtmp-only
```



# pwd

Imprime o nome do diretório atual/trabalho.

Mais informações: <https://www.gnu.org/software/coreutils/pwd>.

- Imprime o diretório atual:

```
pwd
```

- Imprime o diretório atual, e resolve todos os links simbólicos (ou seja, mostra o caminho "físico"):

```
pwd --physical
```

- Imprime o diretório lógico atual:

```
pwd --logical
```

# reboot

Reinicia o sistema.

Mais informações: <https://manned.org/reboot.8>.

- Reinicia o sistema:

```
reboot
```

- Desliga o sistema (igual a `poweroff`):

```
reboot --poweroff
```

- Suspende o sistema (igual a `halt`):

```
reboot --halt
```

- Reinicia imediatamente sem entrar em contato com o gerente do sistema:

```
reboot --force
```

- Escreve a entrada `wtmp shutdown` sem reinicializar o sistema:

```
reboot --wtmp-only
```

# rename

Renomeia múltiplos arquivos.

Nota: essa página refere-se ao comando do pacote **util-linux**.

Para a versão em Perl, veja **file-rename** ou **perl-rename**.

Aviso: Esse comando não tem nenhuma proteção e sobrescreverá arquivos sem aviso prévio.

Mais informações: <https://manned.org/rename>.

- Renomeia arquivos usando substituições simples (substitui 'foo' por 'bar' onde quer que se encontre):

```
rename {{foo}} {{bar}} {{*}}
```

- Dry-run - exibe quais renomeações ocorreriam sem executá-las:

```
rename -vn {{foo}} {{bar}} {{*}}
```

- Não sobrescreve os arquivos existentes:

```
rename -o {{foo}} {{bar}} {{*}}
```

- Altera as extensões dos arquivos:

```
rename {{.ext}} {{.bak}} {{*.ext}}
```

- Acrescenta "foo" no início de todos os nomes de arquivos no diretório atual:

```
rename {{' '}} {{'foo'}} {{*}}
```

- Renomeia um grupo de arquivos com numerações crescente acrescentando zeros aos números até terem 3 dígitos:

```
rename {{foo}} {{foo00}} {{foo?}} && rename {{foo}} {{foo0}}  
{{foo??}}
```

# rm

Remove arquivos ou diretórios.

Veja também: **rmdir**.

Mais informações: <https://www.gnu.org/software/coreutils/rm>.

- Remove arquivos específicos:

```
rm {{caminho/para/arquivo1 caminho/para/arquivo2 ...}}
```

- Remove arquivos específicos ignorando os inexistentes:

```
rm --force {{caminho/para/arquivo1 caminho/para/arquivo2 ...}}
```

- Remove arquivos específicos interativamente avisando antes de cada remoção:

```
rm --interactive {{caminho/para/arquivo1 caminho/para/arquivo2 ...}}
```

- Remove arquivos específicos imprimindo informações sobre cada remoção:

```
rm --verbose {{caminho/para/arquivo1 caminho/para/arquivo2 ...}}
```

- Remove arquivos e diretórios específicos recursivamente:

```
rm --recursive {{caminho/para/arquivo_ou_diretorio1 caminho/para/arquivo_ou_diretorio2 ...}}
```

# rmdir

Remove diretórios sem arquivos.

Veja também: **rm**.

Mais informações: <https://www.gnu.org/software/coreutils/rmdir>.

- Remove diretórios específicos:

```
rmdir {{caminho/para/diretório1 caminho/para/diretório2 ...}}
```

- Remove diretórios aninhados específicos recursivamente:

```
rmdir --parents {{caminho/para/diretório1 caminho/para/diretório2 ...}}
```

# rolldice

Rola dados virtuais.

Mais informações: <https://manned.org/rolldice>.

- Rola um dado de 20 lados:

```
rolldice d{{20}}
```

- Rola dois dados de seis lados e descarta o menor valor:

```
rolldice {{2}}d{{6}}s{{1}}
```

- Rola dois dados de vinte lados e adiciona um modificador ao resultado:

```
rolldice {{2}}d{{20}}{{+5}}
```

- Rola um dado de vinte lados duas vezes:

```
rolldice {{2}}xd{{20}}
```

# sed

Edita texto de uma maneira programável.

Veja também: **awk**, **ed**.

Mais informações: <https://www.gnu.org/software/sed/manual/sed.html>.

- Substitui todas as ocorrências de **apple** (regex básica) por **mango** (regex básica) em todas as linhas de entrada e imprime o resultado na **stdout**:

```
{{comando}} | sed 's/apple/mango/g'
```

- Executa um arquivo de script específico e imprime o resultado na **stdout**:

```
{{comando}} | sed -f {{caminho/para/script.sed}}
```

- Substitui todas as ocorrências de **apple** (regex estendida) por **APPLE** (regex estendida) em todas as linhas de entrada e imprime o resultado na **stdout**:

```
{{comando}} | sed -E 's/(apple)/\U\1/g'
```

- Imprime apenas uma primeira linha na **stdout**:

```
{{comando}} | sed -n '1p'
```

- Substitui todas as ocorrências de **apple** (regex básica) por **mango** (regex básica) em um arquivo específico e sobrescreve o arquivo original no lugar:

```
sed -i 's/apple/mango/g' {{caminho/para/arquivo}}
```

# snapper

Ferramenta de gerenciamento de snapshots do sistema de arquivos.

Mais informações: <http://snapper.io/manpages/snapper.html>.

- Lista configurações de snapshots:

```
snapper list-configs
```

- Cria configuração do snapper:

```
snapper -c {{configuração}} create-config {{caminho/para/diretório}}
```

- Cria um snapshot com uma descrição:

```
snapper -c {{configuração}} create -d  
"{{descrição_do_snapshot}}"
```

- Lista snapshots para uma configuração:

```
snapper -c {{configuração}} list
```

- Exclue um snapshot:

```
snapper -c {{configuração}} delete {{número_do_snapshot}}
```

- Exclue um intervalo de snapshots:

```
snapper -c {{configuração}} delete {{snapshot_X}}-  
{{snapshot_Y}}
```



# systemctl

Controla o sistema systemd e o gerenciador de serviços.

Mais informações: <https://www.freedesktop.org/software/systemd/man/systemctl.html>.

- Mostra todos os serviços em execução:

```
systemctl status
```

- Lista unidades com falha:

```
systemctl --failed
```

- Inicia/Para/Reinicia/Recarrega um serviço:

```
systemctl {{start|stop|restart|reload}} {{unidade}}
```

- Mostra o status de uma unidade:

```
systemctl status {{unidade}}
```

- Ativa/Desativa uma unidade a ser iniciada na inicialização:

```
systemctl {{enable|disable}} {{unidade}}
```

- Mascara/Desmascara uma unidade para impedir ativação e ativação manual:

```
systemctl {{mask|unmask}} {{unidade}}
```

- Recarrega o systemd, verificando por unidades novas ou alteradas:

```
systemctl daemon-reload
```

- Verifica se uma unidades está ativada:

```
systemctl is-enabled {{unidade}}
```

# top

Utilitário para exibir informações, em tempo real, sobre os processos em execução.

Mais informações: <https://manned.org/top>.

- Iniciar top:

```
top
```

- Exibir apenas os processos ativos:

```
top -i
```

- Exibir os processos de um usuário específico:

```
top -u {{username}}
```

- Exibir o(s) processo(s) de um ou mais PID específico(s), separado(s) por vírgula:

```
top -p {{PID1,PID2,PID3}}
```

- Mostra Ajuda sobre os comandos disponíveis:

```
?
```

# ubuntu-bug

Este comando é um pseudônimo de **apport - bug**.

Mais informações: <https://manned.org/ubuntu-bug>.

- Ver documentação sobre o comando original:

```
tldr apport-bug
```

# ufw

Firewall Descomplicado.

Frontend para **iptables** com o objetivo de facilitar a configuração de um firewall.

Mais informações: <https://wiki.ubuntu.com/UncomplicatedFirewall>.

- Habilitar ufw:

```
ufw enable
```

- Desabilitar ufw:

```
ufw disable
```

- Mostrar regras ufw, juntamente com seus números:

```
ufw status numbered
```

- Permitir tráfego de entrada na porta 5432 nesse host com um que identifique o serviço:

```
ufw allow {{5432}} comment "{{Service}}"
```

- Permitir apenas tráfego TCP de 192.168.0.4 pra qualquer endereço deste host, na porta 22 :

```
ufw allow proto {{tcp}} from {{192.168.0.4}} to {{any}} port {{22}}
```

- Negar tráfego na porta 80 desse host :

```
ufw deny {{80}}
```

- Negar todo o tráfego UDP para portas no intervalo 8412:8500:

```
ufw deny proto {{udp}} from {{any}} to {{any}} port {{8412:8500}}
```

- Deletar uma regra particular. O número da regra pode ser recuperado com o `ufw status numbered` comando:

```
ufw delete {{rule_number}}
```

# watch

Executa um comando repetidas vezes, e monitora a saída em tela cheia.

Mais informações: <https://manned.org/watch>.

- Monitora arquivos no diretório atual:

```
watch {{ls}}
```

- Monitora espaço em disco e destaca as alterações:

```
watch -d {{df}}
```

- Monitora processos "node", atualizando a cada 3 segundos:

```
watch -n {{3}} "{{ps aux | grep node}}"
```

# wtf

Mostra a expansão de acrônimos.

Mais informações: <https://manpages.debian.org/latest/bsdgames/wtf.6.en.html>.

- Expande um acrônimo:

```
wtf {{IMO}}
```

- Especifica um tipo de busca computacional:

```
wtf -t {{comp}} {{WWW}}
```

# xwinwrap

Usa um reprodutor de vídeo ou um programa como plano de fundo.

Mais informações: <https://github.com/ujjwal96/xwinwrap>.

- Reproduz um vídeo usando mpv:

```
xwinwrap -b -nf -ov -- {{mpv}} -wid {{wid}} --loop --no-audio  
--no-resume-playback --panscan={{1.0}} {{caminho/para/  
video.mp4}}
```

- Reproduz um vídeo em tela cheia usando mpv:

```
xwinwrap -b -nf -fs -ov -- {{mpv}} -wid {{wid}} --loop --no-  
audio --no-resume-playback --panscan={{1.0}} {{caminho/para/  
video.mp4}}
```

- Reproduz um vídeo usando mpv com 80% de opacidade:

```
xwinwrap -b -nf -ov -o 0.8 --- {{mpv}} -wid {{wid}} --loop --  
no-audio --no-resume-playback --panscan={{1.0}} {{caminho/  
para/video.mp4}}
```

- Reproduz um vídeo usando mpv em um segundo monitor 1600x900 com 1920 de distância do eixo X:

```
xwinwrap -g 1600x900+1920 -b -nf -ov -- {{mpv}} -wid {{wid}}  
--loop --no-audio --no-resume-playback --panscan={{1.0}}  
{{caminho/para/video.mkv}}
```

# yaourt

Utilitário de Arch Linux para compilação de pacotes AUR (Arch User Repository).

Mais informações: <https://linuxcommandlibrary.com/man/yaourt>.

- Sincroniza e atualiza todos os pacotes (incluindo AUR):

```
yaourt -Syua
```

- Instala um novo pacote (incluindo AUR):

```
yaourt -S {{nome_do_pacote}}
```

- Remove um pacote e suas dependências (incluindo pacotes AUR):

```
yaourt -Rs {{nome_do_pacote}}
```

- Procura no banco de dados de pacotes por uma palavra-chave (incluindo AUR):

```
yaourt -Ss {{nome_do_pacote}}
```

- Lista pacotes instalados, versões, e repositórios (pacotes AUR serão listados sob como repositório 'local'):

```
yaourt -Q
```



# yay

Yet Another Yogurt: Um utilitário de Arch Linux para compilar e instalar pacotes do AUR (Arch User Repository).

Veja também **pacman**.

Mais informações: <https://github.com/jguer/yay>.

- Busca interativamente e instala pacotes dos repositórios e AUR:

```
yay {{nome_do_pacote|termo_de_busca}}
```

- Sincroniza e atualiza todos os pacotes dos repositórios e AUR:

```
yay
```

- Sincroniza e atualiza apenas pacotes AUR:

```
yay -Sua
```

- Instala um novo pacote de repositório e AUR:

```
yay -S {{nome_do_pacote}}
```

- Remove um pacote instalado, suas dependências e arquivos de configuração:

```
yay -Rns {{nome_do_pacote}}
```

- Procura no banco de dados de pacotes por uma palavra-chave dos repositórios e AUR:

```
yay -Ss {{palavra_chave}}
```

- Remove pacotes órfãos (instalado como dependência mas não utilizado por qualquer pacote):

```
yay -Yc
```

- Mostra estatísticas dos pacotes instalados e condição do sistema:

```
yay -Ps
```

# yum

Gerenciador de pacotes utilitário para RHEL, Fedora e CentOS (para outras versões).

Mais informações: <https://manned.org/yum>.

- Instala um novo pacote:

```
yum install {{package}}
```

- Instala um novo pacote e assumir sim para todas as questões (também funciona com atualizações, ótimo para atualizações automáticas):

```
yum -y install {{package}}
```

- Localiza o pacote que providência um comando particular:

```
yum provides {{command}}
```

- Remove um pacote:

```
yum remove {{package}}
```

- Exibe atualizações disponíveis para pacotes instalados:

```
yum check-update
```

- Atualiza pacotes instalados para as novas versões disponíveis:

```
yum upgrade
```

# zathura

Um visualizador de documentos modular e baseado em vim, com uma linha de comando integrada.

Tenha certeza de que um backend está instalado (poppler, PostScript, ou DjVu).

Mais informações: <https://pwmt.org/projects/zathura/>.

- Abre um arquivo:

```
zathura {{caminho/para/arquivo}}
```

- Navega esquerda/baixo/cima/direita:

```
<H|J|K|L> ou teclas direcionais
```

- Rotaciona:

```
r
```

- Inverte cores:

```
Ctrl + R
```

- Procura por uma string no documento:

```
/{{string}}
```

- Cria/remove marcadores de página:

```
::{{bmark|bdelete}} {{nome_do_marcador}}
```

- Lista marcadores de página:

```
:blist
```

# zramctl

Configura e controla dispositivos zram.

Use **mkfs** ou **mkswap** para formatar dispositivos zram para partições.

Mais informações: <https://manned.org/zramctl>.

- Verifica se o zram está habilitado:

```
lsmod | grep -i zram
```

- Habilita o zram com um número dinâmico de dispositivos (use `zramctl` para configurar ainda mais os dispositivos):

```
sudo modprobe zram
```

- Habilita o zram com exatamente 2 dispositivos:

```
sudo modprobe zram num_devices={{2}}
```

- Encontra e inicializa o próximo dispositivo zram gratuito em uma unidade virtual de 2 GB usando a compressão LZ4:

```
sudo zramctl --find --size {{2GB}} --algorithm {{lz4}}
```

- Lista dispositivos atualmente inicializados:

```
zramctl
```

# zypper

Utilitário de gerenciamento de pacotes SUSE e openSUSE.

Mais informações: [https://en.opensuse.org/SDB:Zypper\\_manual](https://en.opensuse.org/SDB:Zypper_manual).

- Sincroniza a lista de pacotes e versões disponíveis:

```
zypper refresh
```

- Instala um novo pacote:

```
zypper install {{pacote}}
```

- Remove um pacote:

```
zypper remove {{pacote}}
```

- Atualiza os pacotes instalados para as versões mais recentes disponíveis:

```
zypper update
```

- Pesquisa pacote por palavra-chave:

```
zypper search {{palavra-chave}}
```

- Mostra informações relacionadas aos repositórios configurados:

```
zypper repos --sort-by-priority
```

Osx

# aa

Este comando é um alias de **yaa**.

- Veja a documentação do comando original:

```
tldr yaa
```

# afinfo

Parser de metadados de arquivos de áudio para OS X.

Comando nativo do OS X.

Mais informações: <https://ss64.com/osx/afinfo.html>.

- Exibir informações de um determinado arquivo de áudio:

```
afinfo {{caminho/para/arquivo}}
```

- Imprimir uma descrição de uma linha do arquivo de áudio:

```
afinfo --brief {{caminho/para/arquivo}}
```

- Imprimir informações de metadados e conteúdo do InfoDictionary do arquivo de áudio:

```
afinfo --info {{caminho/para/arquivo}}
```

- Imprimir saída em formato XML:

```
afinfo --xml {{caminho/para/arquivo}}
```

- Imprimir avisos para o arquivo de áudio, se houver:

```
afinfo --warnings {{caminho/para/arquivo}}
```

- Exibir ajuda sobre o uso completo:

```
afinfo --help
```



# afplay

Player de áudio para linha de comando.

Mais informações: <https://ss64.com/osx/afplay.html>.

- Reproduzir um arquivo de som (espera até que a reprodução termine):

```
afplay {{caminho/para/arquivo}}
```

- Reproduzir um arquivo de som em velocidade 2x (taxa de reprodução):

```
afplay --rate {{2}} {{caminho/para/arquivo}}
```

- Reproduzir um arquivo de som em meia velocidade:

```
afplay --rate {{0.5}} {{caminho/para/arquivo}}
```

- Reproduzir os N primeiros segundos de um arquivo de som:

```
afplay --time {{segundos}} {{caminho/para/arquivo}}
```

# airport

Utilitário de configuração de rede sem fio.

Mais informações: <https://ss64.com/osx/airport.html>.

- Mostrar informações de status da rede sem fio atual:

```
airport --getinfo
```

- Farejar tráfego de rede sem fio no canal 1:

```
airport sniff {{1}}
```

- Procurar redes sem fio disponíveis:

```
airport --scan
```

- Desassociar da rede airport atual:

```
sudo airport --disassociate
```

# airportd

Gerencia interfaces sem fio.

Não deve ser invocado manualmente.

Mais informações: <https://www.manpagez.com/man/8/airportd/>.

- Iniciar o daemon:

```
airportd
```

# apachectl

Interface de controle do Servidor HTTP Apache para macOS.

Mais informações: <https://www.unix.com/man-page/osx/8/apachectl/>.

- Iniciar o job launchd org.apache.httpd:

```
apachectl start
```

- Parar o job launchd:

```
apachectl stop
```

- Parar, e então iniciar o job launchd:

```
apachectl restart
```

# applecamerad

Gerenciador de câmera.

Não deve ser invocado manualmente.

Mais informações: <https://www.theiphonewiki.com/wiki/Services>.

- Iniciar o daemon:

```
applecamerad
```

# appsleefd

Fornece serviços app sleep.

Não deve ser invocado manualmente.

Mais informações: <https://keith.github.io/xcode-man-pages/appsleefd.8.html>.

- Iniciar o daemon:

```
appsleefd
```

# arch

Exibe o nome da arquitetura do sistema ou executa um comando em uma arquitetura diferente.

Veja também **uname**.

Mais informações: <https://www.unix.com/man-page/osx/1/arch/>.

- Exibir o nome da arquitetura do sistema:

```
arch
```

- Executar um comando usando a arquitetura x86\_64:

```
arch -x86_64 "{{comando}}"
```

# archey

Ferramenta simples para exibir as informações do sistema com estilo.

Mais informações: <https://github.com/joshfinnie/archey-osx>.

- Mostrar informações do sistema:

```
archey
```

- Mostrar informações do sistema sem saída colorida:

```
archey --nocolor
```

- Mostrar informações do sistema, usando MacPorts em vez de Homebrew:

```
archey --macports
```

- Mostrar informações do sistema sem verificação de endereço IP:

```
archey --offline
```



# as

Montador (assembler) GNU portátil.

Principalmente destinado a montar a saída do **gcc** para ser usada pelo **ld**.

Mais informações: <https://www.unix.com/man-page/osx/1/as/>.

- Montar (compilar) um arquivo, escrevendo a saída para `a.out`:

```
as {{arquivo.s}}
```

- Montar a saída para um determinado arquivo:

```
as {{arquivo.s}} -o {{saida.o}}
```

- Gerar saída mais rapidamente ignorando espaços em branco e pré-processamento de comentários. (Só deve ser usado para compiladores confiáveis):

```
as -f {{arquivo.s}}
```

- Incluir um determinado caminho na lista de diretórios para pesquisar os arquivos especificados nas diretivas `.include`:

```
as -I {{caminho/para/diretório}} {{arquivo.s}}
```

# asr

Restaurar (copiar) uma imagem de disco em um volume.

O nome do comando significa Apple Software Restore.

Mais informações: <https://www.unix.com/man-page/osx/8/asr/>.

- Restaurar uma imagem de disco para um volume de destino:

```
sudo asr restore --source {{nome_da_imagem}}.dmg --target  
{{caminho/para/volume}}
```

- Apagar o volume de destino antes de restaurar:

```
sudo asr restore --source {{nome_da_imagem}}.dmg --target  
{{caminho/para/volume}} --erase
```

- Ignorar a verificação após a restauração:

```
sudo asr restore --source {{nome_da_imagem}}.dmg --target  
{{caminho/para/volume}} --noverify
```

- Clonar volumes sem o uso de uma imagem de disco intermediária:

```
sudo asr restore --source {{caminho/para/volume}} --target  
{{caminho/para/volume_clonado}}
```

# automountd

Um daemon de montagem/desmontagem automática para **autofs**. Iniciado sob demanda por **launchd**.

Não deve ser invocado manualmente.

Mais informações: <https://www.manpagez.com/man/8/automountd/>.

- Iniciar o daemon:

```
automountd
```

- Log de mais detalhes em syslog:

```
automountd -v
```

# avbdeviced

Serviço para gerenciar dispositivos Audio Video Bridging (AVB).

Não deve ser invocado manualmente.

Mais informações: <https://www.manpagez.com/man/1/avbdeviced/>.

- Iniciar o daemon:

```
avbdeviced
```

# base64

Codifica e decodifica usando a representação Base64.

Mais informações: <https://www.unix.com/man-page/osx/1/base64/>.

- Codificar um arquivo:

```
base64 --input={{arquivo}}
```

- Decodificar um arquivo:

```
base64 --decode --input={{arquivo_base64}}
```

- Codificar de stdin:

```
echo -n "{{texto}}" | base64
```

- Decodificar de stdin:

```
echo -n {{texto_base64}} | base64 --decode
```

# bc

Linguagem e calculadora com precisão arbitrária.

Veja também: **dc**.

Mais informações: <https://manned.org/man/freebsd-13.0/bc.1>.

- Iniciar uma sessão interativa:

```
bc
```

- Iniciar uma sessão interativa com a biblioteca matemática padrão habilitada:

```
bc --mathlib
```

- Calcular uma expressão:

```
bc --expression='{{5 / 3}}'
```

- Executar um script:

```
bc {{caminho/para/script.bc}}
```

- Calcular uma expressão com a escala especificada:

```
bc --expression='scale = {{10}}; {{5 / 3}}'
```

- Calcular uma função sine/cosine/arctangent/natural logarithm/exponential usando `mathlib`:

```
bc --mathlib --expression='{{s|c|a|l|e}}({{1}})'
```

# bird

Suporta a sincronização do iCloud e iCloud Drive.

Não deve ser invocado manualmente.

Mais informações: <https://www.unix.com/man-page/mojave/8/bird/>.

- Iniciar o daemon:

```
bird
```

# bless

Define a capacidade de inicialização por volume e as opções de disco de inicialização. Set volume boot capability and startup disk options.

Mais informações: <https://ss64.com/osx/bless.html>.

- Definir um volume somente com Mac OS X ou Darwin, e criar os arquivos BootX e `boot.efi` se necessário:

```
bless --folder {{/Volumes/Mac OS X/System/Library/
CoreServices}} --bootinfo --bootefi
```

- Definir um volume contendo Mac OS 9 ou Mac OS X como o volume ativo:

```
bless --mount {{/Volumes/Mac OS}} --setBoot
```

- Definir o sistema para NetBoot e transmitir para um servidor disponível:

```
bless --netboot --server {{bsdp://255.255.255.255}}
```

- Coletar informações sobre o volume atualmente selecionado (conforme determinado pelo firmware), adequado para piping para um programa capaz de analisar listas de propriedades:

```
bless --info --plist
```



# bnepd

Serviço que lida com todas as conexões de rede Bluetooth.

Não deve ser invocado manualmente.

Mais informações: <https://www.unix.com/man-page/osx/8/bnepd/>.

- Iniciar o daemon:

```
bnepd
```

# brightness

Obtém e define o nível de brilho de todos os monitores internos e alguns monitores externos.

Mais informações: <https://github.com/nriley/brightness>.

- Mostrar o brilho atual:

```
brightness -l
```

- Definir o brilho para 100%:

```
brightness {{1}}
```

- Definir o brilho para 50%:

```
brightness {{0.5}}
```

# caffeinate

Evita que o macOS entre em suspensão (repouso).

Mais informações: <https://ss64.com/osx/caffeinate.html>.

- Evita a suspensão por uma hora (3600 segundos):

```
caffeinate -u -t {{3600}}
```

- Evita a suspensão até que um comando seja concluído:

```
caffeinate -s "{{comando}}"
```

- Evita a suspensão até que você digite Ctrl-C:

```
caffeinate -i
```

# cal

Exibe informações de calendário.

Mais informações: <https://ss64.com/osx/cal.html>.

- Exibe um calendário para o mês atual:

```
cal
```

- Exibe os meses anterior, atual, e próximo:

```
cal -3
```

- Exibe um calendário para um mês específico (1-12 ou nome):

```
cal -m {{mês}}
```

- Exibe um calendário para o ano atual:

```
cal -y
```

- Exibe um calendário para um ano específico (4 dígitos):

```
cal {{ano}}
```

- Exibe um calendário para um mês e ano específicos:

```
cal {{mês}} {{ano}}
```

- Exibe a data da Páscoa (igrejas cristãs ocidentais) em um determinado ano:

```
ncal -e {{ano}}
```

# carthage

Ferramenta de gerenciamento de dependências para aplicativos Cocoa.

Mais informações: <https://github.com/Carthage/Carthage>.

- Baixa a versão mais recente de todas as dependências mencionadas no Cartfile e realiza o build delas:

```
carthage update
```

- Atualiza as dependências, e faz build apenas para o iOS:

```
carthage update --platform ios
```

- Atualiza as dependências, sem realizar build de nenhuma delas:

```
carthage update --no-build
```

- Faz o download e rebuild da versão atual das dependências (sem atualizá-las):

```
carthage bootstrap
```

- Faz o rebuild de uma dependência específica:

```
carthage build {{dependência}}
```

# cfprefsd

Fornece serviços de preferências (**CFPreferences**, **NSUserDefaults**).

Não deve ser invocado manualmente.

Mais informações: <https://www.unix.com/man-page/osx/8/cfprefsd/>.

- Inicia o daemon:

```
cfprefsd
```

# chflags

Altera flags de arquivo ou diretório.

Mais informações: <https://ss64.com/osx/chflags.html>.

- Define a flag `hidden` para um arquivo:

```
chflags {{hidden}} {{caminho/para/arquivo}}
```

- Remove a flag `hidden` de um arquivo:

```
chflags {{nohidden}} {{caminho/para/arquivo}}
```

- Define recursivamente a flag `uchg` para um diretório:

```
chflags -R {{uchg}} {{caminho/para/diretório}}
```

- Remove recursivamente a flag `uchg` de um diretório:

```
chflags -R {{nouchg}} {{caminho/para/diretório}}
```

# cloudphotod

Sincroniza fotos do iCloud.

Não deve ser invocado manualmente.

Mais informações: <https://www.manpagez.com/man/8/cloudphotosd/>.

- Inicia o daemon:

```
cloudphotod
```



# codesign

Cria e manipula assinaturas de código para macOS.

Mais informações: <https://www.unix.com/man-page/osx/1/codesign/>.

- Assina um aplicativo com um certificado:

```
codesign --sign "{{Nome da Minha Empresa}}" {{caminho/para/  
App.app}}
```

- Verifica o certificado de um aplicativo:

```
codesign --verify {{caminho/para/App.app}}
```

# coreaudiod

Serviço para o Core Audio, o sistema de áudio da Apple.

Não deve ser invocado manualmente.

Mais informações: <https://developer.apple.com/library/archive/documentation/MusicAudio/Conceptual/CoreAudioOverview/WhatIsCoreAudio/WhatIsCoreAudio.html>.

- Inicia o daemon:

```
coreaudiod
```

# coreautha

Um agente de sistema que fornece o framework **LocalAuthentication**.

Não deve ser invocado manualmente. Veja também: **coreauthd**.

Mais informações: <https://www.manpagez.com/man/8/coreautha/>.

- Inicia o agente:

`coreautha`

# coreauthd

Um daemon de sistema que fornece o framework **LocalAuthentication**.

Não deve ser invocado manualmente. Veja também: **coreauthd**.

Mais informações: <https://www.manpagez.com/man/8/coreauthd/>.

- Inicia o daemon:

`coreauthd`

# cot

Editor de texto puro para macOS.

Mais informações: <https://coteditor.com/>.

- Inicia o CotEditor:

```
cot
```

- Abre arquivos específicos:

```
cot {{caminho/para/arquivo1 caminho/para/arquivo2 ...}}
```

- Abre um novo documento em branco:

```
cot --new
```

- Abre um arquivo específico e bloqueia o terminal até que o arquivo seja fechado:

```
cot --wait {{caminho/para/arquivo}}
```

- Abre um arquivo específico com o cursor em uma linha e coluna especificada:

```
cot --line {{número_da_linha}} --column {{número_da_coluna}}  
{{caminho/para/arquivo}}
```

# csrutil

Gerencia a configuração do System Integrity Protection (SIP).

Mais informações: <https://ss64.com/osx/csrutil.html>.

- Exibe o status do System Integrity Protection:

```
csrutil status
```

- Desabilita o System Integrity Protection:

```
csrutil disable
```

- Habilita o System Integrity Protection:

```
csrutil enable
```

- Exibe a lista de origens permitidas do NetBoot:

```
csrutil netboot list
```

- Adiciona um endereço IPv4 à lista de origens permitidas do NetBoot:

```
csrutil netboot add {{endereço_ip}}
```

- Reseta o status do System Integrity Protection e limpa a lista do NetBoot:

```
csrutil clear
```

# csshX

Ferramenta de Cluster SSH para macOS.

Mais informações: <https://github.com/brockgr/csshx>.

- Conecta a vários hosts:

```
csshX {{nomedohost1}} {{nomedohost2}}
```

- Conecta a vários hosts com uma determinada chave SSH:

```
csshX {{user@nomedohost1}} {{user@nomedohost2}} --ssh_args "-  
i {{caminho/para/ssh_key.pem}}"
```

- Conecta a um cluster predefinido em `/etc/clusters`:

```
csshX cluster1
```

# cut

Recorta campos de **stdin** ou arquivos.

Mais informações: <https://manned.org/man/freebsd-13.0/cut.1>.

- Imprime um intervalo específico de caracteres/campos de cada linha:

```
{{comando}} | cut -{{c|f}} {{1|1,10|1-10|1-|-10}}
```

- Imprime um intervalo de cada linha com um delimitador específico:

```
{{comando}} | cut -d "{{{,}}}" -{{c}} {{1}}
```

- Imprime um intervalo de cada linha de um arquivo específico:

```
cut -{{c}} {{1}} {{caminho/para/arquivo}}
```



# dark-mode

Controla o modo escuro do macOS a partir da linha de comando.

Mais informações: <https://github.com/sindresorhus/dark-mode>.

- Alterna o modo escuro (ativa se estiver desativado, desativa se estiver ativado):

```
dark-mode
```

- Ativa o modo escuro:

```
dark-mode on
```

- Desativa o modo escuro:

```
dark-mode off
```

- Verifica se o modo escuro está ativado:

```
dark-mode status
```

# date

Define ou exibe a data do sistema.

Mais informações: <https://ss64.com/osx/date.html>.

- Exibe a data atual usando o formato da localidade padrão:

```
date +%c
```

- Exibe a data atual no formato UTC e ISO 8601:

```
date -u +%Y-%m-%dT%H:%M:%SZ
```

- Exibe a data atual como um timestamp Unix (segundos desde a época Unix):

```
date +%s
```

- Exibe uma data específica (representada como um timestamp Unix) usando o formato padrão:

```
date -r 1473305798
```

# dd

Converte e copia um arquivo.

Mais informações: <https://keith.github.io/xcode-man-pages/dd.1.html>.

- Cria uma unidade USB inicializável a partir de um arquivo isohybrid (tal como `archlinux-xxx.iso`):

```
dd if={{arquivo.iso}} of=/dev/{{unidade_usb}}
```

- Clona uma unidade para outra unidade com bloco de 4 MB e ignora erro:

```
dd if=/dev/{{unidade_origem}} of=/dev/{{unidade_destino}}  
bs=4m conv=noerror
```

- Gera um arquivo de 100 bytes aleatórios usando o driver aleatório do kernel:

```
dd if=/dev/urandom of={{arquivo_aleatório}} bs=100 count=1
```

- Compara o desempenho de gravação de um disco:

```
dd if=/dev/zero of={{arquivo_1GB}} bs=1024 count=1000000
```

# defaults

Lê e grava a configuração do usuário do macOS para aplicativos.

Mais informações: <https://ss64.com/osx/defaults.html>.

- Lê os padrões do sistema para uma opção do aplicativo:

```
defaults read "{{aplicativo}}" "{{opção}}"
```

- Lê os valores padrão para uma opção do aplicativo:

```
defaults read -app "{{aplicativo}}" "{{opção}}"
```

- Pesquisa uma palavra-chave em nomes de domínio, chaves, e valores:

```
defaults find "{{palavra-chave}}"
```

- Grava o valor padrão de uma opção do aplicativo:

```
defaults write "{{aplicativo}}" "{{opção}}" {{-tipo}}  
{{valor}}
```

- Acelera as animações do Mission Control:

```
defaults write com.apple.Dock expose-animation-duration -  
float 0.1
```

- Exclui todos os padrões de um aplicativo:

```
defaults delete "{{aplicativo}}"
```

# deleted

Acompanha o espaço purgável e solicita que os clientes removam os arquivos quando o espaço estiver baixo.

Não deve ser invocado manualmente.

Mais informações: <https://www.manpagez.com/man/8/deleted/>.

- Inicia o daemon:

`deleted`

# dhcp6d

Servidor DHCPv6 stateless. Veja também: **InternetSharing**.

Não deve ser invocado manualmente.

Mais informações: <https://www.manpagez.com/man/8/dhcp6d/>.

- Inicia o daemon:

```
dhcp6d
```

- Usa uma configuração personalizada:

```
dhcp6d {{caminho/para/configuração}}
```

# diskutil

Utilitário para gerenciar discos e volumes locais.

Mais informações: <https://ss64.com/osx/diskutil.html>.

- Lista todos os discos, partições, e volumes montados atualmente disponíveis:

```
diskutil list
```

- Repara as estruturas de dados do sistema de arquivos de um volume:

```
diskutil repairVolume {{/dev/diskX}}
```

- Desmonta um volume:

```
diskutil unmountDisk {{/dev/diskX}}
```

- Ejeta um CD/DVD (desmonta primeiro):

```
diskutil eject {{/dev/disk1}}
```

# distnoted

Fornece serviços de notificação distribuídos.

Não deve ser invocado manualmente.

Mais informações: <https://www.manpagez.com/man/8/distnoted/>.

- Inicia o daemon:

```
distnoted
```



# ditto

Copia arquivos e diretórios.

Mais informações: <https://ss64.com/osx/ditto.html>.

- Sobrescreve o conteúdo do diretório de destino pelo conteúdo do diretório de origem:

```
ditto {{caminho/para/origem}} {{caminho/para/destino}}
```

- Imprime uma linha na janela do Terminal para cada arquivo que está sendo copiado:

```
ditto -V {{caminho/para/origem}} {{caminho/para/destino}}
```

- Copia um determinado arquivo ou diretório, mantendo as permissões do arquivo original:

```
ditto -rsrc {{caminho/para/origem}} {{caminho/para/destino}}
```

# dmesg

Exibe mensagens do kernel na saída padrão.

Mais informações: <https://www.manpagez.com/man/8/dmesg/>.

- Exibe mensagens do kernel:

```
dmesg
```

- Exibe quanta memória física está disponível no sistema:

```
dmesg | grep -i memory
```

- Exibe mensagens do kernel, 1 página por vez:

```
dmesg | less
```

# dot\_clean

Mescla `.*` arquivos com arquivos nativos correspondentes.

Mais informações: [https://ss64.com/osx/dot\\_clean.html](https://ss64.com/osx/dot_clean.html).

- Mescla todos os `.*` arquivos recursivamente:

```
dot_clean {{caminho/para/diretório}}
```

- Não mescla recursivamente todos `.*` em um diretório (flat merge):

```
dot_clean -f {{caminho/para/diretório}}
```

- Mescla e exclui todos os arquivos `.*`:

```
dot_clean -m {{caminho/para/diretório}}
```

- Somente exclui arquivos `.*` se houver um arquivo nativo correspondente:

```
dot_clean -n {{caminho/para/diretório}}
```

- Segue os links simbólicos:

```
dot_clean -s {{caminho/para/diretório}}
```

- Imprime saída verbosa:

```
dot_clean -v {{caminho/para/diretório}}
```

# drutil

Interage com gravadores de DVD.

Mais informações: <https://ss64.com/osx/drutil.html>.

- Ejeta um disco da unidade:

```
drutil eject
```

- Grava um diretório como um sistema de arquivos ISO9660 em um DVD. Não verifica, e ejeta quando terminar:

```
drutil burn -noverify -eject -iso9660
```

# du

Uso do Disco: estima e resume o uso do espaço de arquivos e diretórios.

Mais informações: <https://ss64.com/osx/du.html>.

- Lista os tamanhos de um diretório e quaisquer subdiretórios, na unidade fornecida (KiB/MiB/GiB):

```
du -{{k|m|g}} {{caminho/para/diretório}}
```

- Lista os tamanhos de um diretório e quaisquer subdiretórios, em formato legível (ou seja, selecionando automaticamente a unidade apropriada para cada tamanho):

```
du -h {{caminho/para/diretório}}
```

- Exibe o tamanho de um único diretório, em unidades legíveis:

```
du -sh {{caminho/para/diretório}}
```

- Lista os tamanhos legíveis de um diretório e de todos os arquivos e diretórios dentro dele:

```
du -ah {{caminho/para/diretório}}
```

- Lista os tamanhos legíveis de um diretório e quaisquer subdiretórios, até N níveis de profundidade:

```
du -h -d {{N}} {{caminho/para/diretório}}
```

- Lista o tamanho legível de todos os arquivos .jpg nos subdiretórios do diretório atual e exibe um total cumulativo no final:

```
du -ch {{*/*.jpg}}
```

# duti

Define os aplicativos padrão para tipos de documentos e esquemas de URL no macOS.

Mais informações: <https://github.com/moretension/duti>.

- Define o Safari como o manipulador padrão de documentos HTML:

```
duti -s {{com.apple.Safari}} {{public.html}} all
```

- Define o VLC como visualizador padrão para arquivos com extensões .m4v:

```
duti -s {{org.videolan.vlc}} {{m4v}} viewer
```

- Define o Finder como o manipulador padrão para esquema de URL ftp://:

```
duti -s {{com.apple.Finder}} "{{ftp}}"
```

- Exibe informações sobre o aplicativo padrão para uma determinada extensão:

```
duti -x {{ext}}
```

- Exibe o manipulador padrão para um determinado UTI:

```
duti -d {{uti}}
```

- Exibe todos os manipuladores de um determinado UTI:

```
duti -l {{uti}}
```

# ed

Editor de texto original do Unix.

Veja também: **awk**, **sed**.

Mais informações: [https://www.gnu.org/software/ed/manual/ed\\_manual.html](https://www.gnu.org/software/ed/manual/ed_manual.html).

- Inicia uma sessão interativa do editor com um documento vazio:

```
ed
```

- Inicia uma sessão interativa do editor com um documento vazio e um [p]rompt específico:

```
ed -p '> '
```

- Inicia uma sessão interativa do editor com um documento vazio e sem diagnósticos, contagens de bytes, e prompt '!':

```
ed -s
```

- Edita um arquivo específico (mostra a contagem de bytes do arquivo carregado):

```
ed {{caminho/para/arquivo}}
```

- Substitui uma string por uma substituição específica em todas as linhas:

```
,s/{{expressão_regular}}/{{substituição}}/g
```

# emond

Serviço Event Monitor que aceita eventos de vários serviços, os executa por meio de um mecanismo de regras simples, e executa ações.

As ações podem executar comandos, enviar e-mails, ou mensagens SMS.

Mais informações: <https://www.manpagez.com/man/8/emond/>.

- Inicia o daemon:

```
emond
```

- Especifica as regras para o emond processar, fornecendo um caminho para um arquivo ou diretório:

```
emond -r {{caminho/para/arquivo_ou_diretório}}
```

- Usa um arquivo de configuração específico:

```
emond -c {{caminho/para/configuração}}
```



# fdesetup

Define e recupera informações relacionadas ao FileVault.

Mais informações: <https://www.unix.com/man-page/mojave/8/fdesetup/>.

- Lista os usuários atuais habilitados para o FileVault:

```
sudo fdesetup list
```

- Obtém o status atual do FileVault:

```
fdesetup status
```

- Adiciona usuário habilitado para o FileVault:

```
sudo fdesetup add -usertoadd user1
```

- Ativa o FileVault:

```
sudo fdesetup enable
```

- Desativa o FileVault:

```
sudo fdesetup disable
```

# filecoordinationd

Coordena o acesso a arquivos por vários processos (**NSFileCoordinator**, **NSFilePresenter**).

Não deve ser invocado manualmente.

Mais informações: <https://www.unix.com/man-page/osx/8/filecoordinationd/>.

- Inicia o daemon:

```
filecoordinationd
```

# fileicon

Uma CLI do macOS para gerenciar ícones personalizados de arquivos e pastas.

Mais informações: <https://github.com/mklement0/fileicon>.

- Define um ícone personalizado para um arquivo ou diretório específico:

```
fileicon set {{caminho/para/arquivo_ou_diretório}} {{caminho/para/icone.png}}
```

- Remove um ícone personalizado de um arquivo ou diretório específico:

```
fileicon rm {{caminho/para/arquivo_ou_diretório}}
```

- Salva o ícone personalizado de um arquivo ou diretório como um arquivo `.icns` no diretório atual:

```
fileicon get {{caminho/para/arquivo_ou_diretório}}
```

- Testa se um arquivo ou diretório específico tem um ícone personalizado:

```
fileicon test {{caminho/para/arquivo_ou_diretório}}
```

# fontd

Disponibiliza fontes para o sistema.

Não deve ser invocado manualmente.

Mais informações: <https://www.manpagez.com/man/8/fontd/>.

- Inicia o daemon:

```
fontd
```

# fsck

Verifica a integridade de um sistema de arquivos ou repara ele. O sistema de arquivos deve ser desmontado no momento em que o comando é executado.

É um wrapper que chama **fsck\_hfs**, **fsck\_apfs**, **fsck\_msdos**, **fsck\_exfat**, e **fsck\_udf** conforme necessário.

Mais informações: <https://ss64.com/osx/fsck.html>.

- Verifica o sistema de arquivos `/dev/sdX`, relatando quaisquer blocos danificados:

```
fsck {{/dev/sdX}}
```

- Verifica o sistema de arquivos `/dev/sdX` apenas se estiver limpo, relatando quaisquer blocos danificados e permitindo que o usuário interativamente escolha reparar cada um deles:

```
fsck -f {{/dev/sdX}}
```

- Verifica o sistema de arquivos `/dev/sdX` apenas se estiver limpo, relatando quaisquer blocos danificados e reparando-os automaticamente:

```
fsck -fy {{/dev/sdX}}
```

- Verifica o sistema de arquivos `/dev/sdX`, informando se ele foi desmontado corretamente:

```
fsck -q {{/dev/sdX}}
```

# g[

Este comando é um pseudônimo de **-p linux** [.

- Ver documentação sobre o comando original:

```
tldr -p linux [
```

# gawk

Este comando é um pseudônimo de **-p linux awk**.

- Ver documentação sobre o comando original:

```
tldr -p linux awk
```

# gb2sum

Este comando é um pseudônimo de **-p linux b2sum**.

- Ver documentação sobre o comando original:

```
tldr -p linux b2sum
```



# gbase32

Este comando é um pseudônimo de **-p linux base32**.

- Ver documentação sobre o comando original:

```
tldr -p linux base32
```

# gbase64

Este comando é um pseudônimo de **-p linux base64**.

- Ver documentação sobre o comando original:

```
tldr -p linux base64
```

# gbasename

Este comando é um pseudônimo de **-p linux basename**.

- Ver documentação sobre o comando original:

```
tldr -p linux basename
```

# gbasenc

Este comando é um pseudônimo de **-p linux basenc**.

- Ver documentação sobre o comando original:

```
tldr -p linux basenc
```

# gcat

Este comando é um pseudônimo de **-p linux cat**.

- Ver documentação sobre o comando original:

```
tldr -p linux cat
```

# gchcon

Este comando é um pseudônimo de **-p linux chcon**.

- Ver documentação sobre o comando original:

```
tldr -p linux chcon
```

# gchgrp

Este comando é um pseudônimo de **-p linux chgrp**.

- Ver documentação sobre o comando original:

```
tldr -p linux chgrp
```

# gchmod

Este comando é um pseudônimo de **-p linux chmod**.

- Ver documentação sobre o comando original:

```
tldr -p linux chmod
```



# gchown

Este comando é um pseudônimo de **-p linux chown**.

- Ver documentação sobre o comando original:

```
tldr -p linux chown
```

# gchroot

Este comando é um pseudônimo de **-p linux chroot**.

- Ver documentação sobre o comando original:

```
tldr -p linux chroot
```

# gcksum

Este comando é um pseudônimo de **-p linux cksum**.

- Ver documentação sobre o comando original:

```
tldr -p linux cksum
```

# gcomm

Este comando é um pseudônimo de **-p linux comm**.

- Ver documentação sobre o comando original:

```
tldr -p linux comm
```

# gcp

Este comando é um pseudônimo de **-p linux cp**.

- Ver documentação sobre o comando original:

```
tldr -p linux cp
```

# gcsplit

Este comando é um pseudônimo de **-p linux csplit**.

- Ver documentação sobre o comando original:

```
tldr -p linux csplit
```

# gcut

Este comando é um pseudônimo de **-p linux cut**.

- Ver documentação sobre o comando original:

```
tldr -p linux cut
```

# gdate

Este comando é um pseudônimo de **-p linux date**.

- Ver documentação sobre o comando original:

```
tldr -p linux date
```



# gdd

Este comando é um pseudônimo de **-p linux dd**.

- Ver documentação sobre o comando original:

```
tldr -p linux dd
```

# gdf

Este comando é um pseudônimo de **-p linux df**.

- Ver documentação sobre o comando original:

```
tldr -p linux df
```

# gdir

Este comando é um pseudônimo de **-p linux dir**.

- Ver documentação sobre o comando original:

```
tldr -p linux dir
```

# gdircolors

Este comando é um pseudônimo de **-p linux dircolors**.

- Ver documentação sobre o comando original:

```
tldr -p linux dircolors
```

# gdirname

Este comando é um pseudônimo de **-p linux dirname**.

- Ver documentação sobre o comando original:

```
tldr -p linux dirname
```

# gdnssdomainname

Este comando é um pseudônimo de **-p linux dnssdomainname**.

- Ver documentação sobre o comando original:

```
tldr -p linux dnssdomainname
```

# gecho

Este comando é um pseudônimo de **-p linux echo**.

- Ver documentação sobre o comando original:

```
tldr -p linux echo
```

# ged

Este comando é um pseudônimo de **-p linux ed**.

- Ver documentação sobre o comando original:

```
tldr -p linux ed
```



# gegrep

Este comando é um pseudônimo de **-p linux egrep**.

- Ver documentação sobre o comando original:

```
tldr -p linux egrep
```

# genv

Este comando é um pseudônimo de **-p linux env**.

- Ver documentação sobre o comando original:

```
tldr -p linux env
```

# GetFileInfo

Obtém informações sobre um arquivo em um diretório HFS+.

Mais informações: <https://www.unix.com/man-page/osx/1/GetFileInfo/>.

- Exibe informações sobre um determinado arquivo:

```
GetFileInfo {{caminho/para/nome_do_arquivo}}
```

- Exibe a data e hora em que um determinado arquivo foi criado:

```
GetFileInfo -d {{caminho/para/nome_do_arquivo}}
```

- Exibe a data e hora em que um determinado arquivo foi modificado pela última vez:

```
GetFileInfo -m {{caminho/para/nome_do_arquivo}}
```

- Exibe o criador de um determinado arquivo:

```
GetFileInfo -c {{caminho/para/nome_do_arquivo}}
```

# gexpand

Este comando é um pseudônimo de **-p linux expand**.

- Ver documentação sobre o comando original:

```
tldr -p linux expand
```

# gexpr

Este comando é um pseudônimo de **-p linux expr**.

- Ver documentação sobre o comando original:

```
tldr -p linux expr
```

# gfactor

Este comando é um pseudônimo de **-p linux factor**.

- Ver documentação sobre o comando original:

```
tldr -p linux factor
```

# gfalse

Este comando é um pseudônimo de **-p linux false**.

- Ver documentação sobre o comando original:

```
tldr -p linux false
```

# gfgrep

Este comando é um pseudônimo de **-p linux fgrep**.

- Ver documentação sobre o comando original:

```
tldr -p linux fgrep
```



# gfind

Este comando é um pseudônimo de **-p linux find**.

- Ver documentação sobre o comando original:

```
tldr -p linux find
```

# gfmt

Este comando é um pseudônimo de **-p linux fmt**.

- Ver documentação sobre o comando original:

```
tldr -p linux fmt
```

# gfold

Este comando é um pseudônimo de **-p linux fold**.

- Ver documentação sobre o comando original:

```
tldr -p linux fold
```

# gftp

Este comando é um pseudônimo de **-p linux ftp**.

- Ver documentação sobre o comando original:

```
tldr -p linux ftp
```

# ggrep

Este comando é um pseudônimo de **-p linux grep**.

- Ver documentação sobre o comando original:

```
tldr -p linux grep
```

# ggroups

Este comando é um pseudônimo de **-p linux groups**.

- Ver documentação sobre o comando original:

```
tldr -p linux groups
```

# ghead

Este comando é um pseudônimo de **-p linux head**.

- Ver documentação sobre o comando original:

```
tldr -p linux head
```

# ghostid

Este comando é um pseudônimo de **-p linux hostid**.

- Ver documentação sobre o comando original:

```
tldr -p linux hostid
```



# ghostname

Este comando é um pseudônimo de **-p linux hostname**.

- Ver documentação sobre o comando original:

```
tldr -p linux hostname
```

# gid

Este comando é um pseudônimo de **-p linux id**.

- Ver documentação sobre o comando original:

```
tldr -p linux id
```

# gifconfig

Este comando é um pseudônimo de **-p linux ifconfig**.

- Ver documentação sobre o comando original:

```
tldr -p linux ifconfig
```

# gindent

Este comando é um pseudônimo de **-p linux indent**.

- Ver documentação sobre o comando original:

```
tldr -p linux indent
```

# ginstall

Este comando é um pseudônimo de **-p linux install**.

- Ver documentação sobre o comando original:

```
tldr -p linux install
```

# gjoin

Este comando é um pseudônimo de **-p linux join**.

- Ver documentação sobre o comando original:

```
tldr -p linux join
```

# gkill

Este comando é um pseudônimo de **-p linux kill**.

- Ver documentação sobre o comando original:

```
tldr -p linux kill
```

# glibtool

Este comando é um pseudônimo de **-p linux libtool**.

- Ver documentação sobre o comando original:

```
tldr -p linux libtool
```



# glibtoolize

Este comando é um pseudônimo de **-p linux libtoolize**.

- Ver documentação sobre o comando original:

```
tldr -p linux libtoolize
```

# glink

Este comando é um pseudônimo de **-p linux link**.

- Ver documentação sobre o comando original:

```
tldr -p linux link
```

# gln

Este comando é um pseudônimo de **-p linux ln**.

- Ver documentação sobre o comando original:

```
tldr -p linux ln
```

# glocate

Este comando é um pseudônimo de **-p linux locate**.

- Ver documentação sobre o comando original:

```
tldr -p linux locate
```

# glogger

Este comando é um pseudônimo de **-p linux logger**.

- Ver documentação sobre o comando original:

```
tldr -p linux logger
```

# glogname

Este comando é um pseudônimo de **-p linux logname**.

- Ver documentação sobre o comando original:

```
tldr -p linux logname
```

# gls

Este comando é um pseudônimo de **-p linux ls**.

- Ver documentação sobre o comando original:

```
tldr -p linux ls
```

# gmake

Este comando é um pseudônimo de **-p linux make**.

- Ver documentação sobre o comando original:

```
tldr -p linux make
```



# gmd5sum

Este comando é um pseudônimo de **-p linux md5sum**.

- Ver documentação sobre o comando original:

```
tldr -p linux md5sum
```

# gmkdir

Este comando é um pseudônimo de **-p linux mkdir**.

- Ver documentação sobre o comando original:

```
tldr -p linux mkdir
```

# gmkfifo

Este comando é um pseudônimo de **-p linux mkfifo**.

- Ver documentação sobre o comando original:

```
tldr -p linux mkfifo
```

# gmknod

Este comando é um pseudônimo de **-p linux mknod**.

- Ver documentação sobre o comando original:

```
tldr -p linux mknod
```

# gmkttemp

Este comando é um pseudônimo de **-p linux mktemp**.

- Ver documentação sobre o comando original:

```
tldr -p linux mktemp
```

# gmv

Este comando é um pseudônimo de **-p linux mv**.

- Ver documentação sobre o comando original:

```
tldr -p linux mv
```

# gnice

Este comando é um pseudônimo de **-p linux nice**.

- Ver documentação sobre o comando original:

```
tldr -p linux nice
```

# gnl

Este comando é um pseudônimo de **-p linux nl**.

- Ver documentação sobre o comando original:

```
tldr -p linux nl
```



# gnohup

Este comando é um pseudônimo de **-p linux nohup**.

- Ver documentação sobre o comando original:

```
tldr -p linux nohup
```

# gnproc

Este comando é um pseudônimo de **-p linux nproc**.

- Ver documentação sobre o comando original:

```
tldr -p linux nproc
```

# gnumfmt

Este comando é um pseudônimo de **-p linux numfmt**.

- Ver documentação sobre o comando original:

```
tldr -p linux numfmt
```

# god

Este comando é um pseudônimo de **-p linux od**.

- Ver documentação sobre o comando original:

```
tldr -p linux od
```

# gpaste

Este comando é um pseudônimo de **-p linux paste**.

- Ver documentação sobre o comando original:

```
tldr -p linux paste
```

# gpathchk

Este comando é um pseudônimo de **-p linux pathchk**.

- Ver documentação sobre o comando original:

```
tldr -p linux pathchk
```

# gping

Este comando é um pseudônimo de **-p linux ping**.

- Ver documentação sobre o comando original:

```
tldr -p linux ping
```

# gping6

Este comando é um pseudônimo de **-p linux ping6**.

- Ver documentação sobre o comando original:

```
tldr -p linux ping6
```



# gpinky

Este comando é um pseudônimo de **-p linux pinky**.

- Ver documentação sobre o comando original:

```
tldr -p linux pinky
```

# gpr

Este comando é um pseudônimo de **-p linux pr**.

- Ver documentação sobre o comando original:

```
tldr -p linux pr
```

# gprintenv

Este comando é um pseudônimo de **-p linux printenv**.

- Ver documentação sobre o comando original:

```
tldr -p linux printenv
```

# gprintf

Este comando é um pseudônimo de **-p linux printf**.

- Ver documentação sobre o comando original:

```
tldr -p linux printf
```

# gptx

Este comando é um pseudônimo de **-p linux ptx**.

- Ver documentação sobre o comando original:

```
tldr -p linux ptx
```

# gpwd

Este comando é um pseudônimo de **-p linux pwd**.

- Ver documentação sobre o comando original:

```
tldr -p linux pwd
```

# grcp

Este comando é um pseudônimo de **-p linux rcp**.

- Ver documentação sobre o comando original:

```
tldr -p linux rcp
```

# greadlink

Este comando é um pseudônimo de **-p linux readlink**.

- Ver documentação sobre o comando original:

```
tldr -p linux readlink
```



# grealpath

Este comando é um pseudônimo de **-p linux realpath**.

- Ver documentação sobre o comando original:

```
tldr -p linux realpath
```

# grexec

Este comando é um pseudônimo de **-p linux rexec**.

- Ver documentação sobre o comando original:

```
tldr -p linux rexec
```

# grlogin

Este comando é um pseudônimo de **-p linux rlogin**.

- Ver documentação sobre o comando original:

```
tldr -p linux rlogin
```

# grm

Este comando é um pseudônimo de **-p linux rm**.

- Ver documentação sobre o comando original:

```
tldr -p linux rm
```

# grmdir

Este comando é um pseudônimo de **-p linux rmdir**.

- Ver documentação sobre o comando original:

```
tldr -p linux rmdir
```

# grsh

Este comando é um pseudônimo de **-p linux rsh**.

- Ver documentação sobre o comando original:

```
tldr -p linux rsh
```

# gruncon

Este comando é um pseudônimo de **-p linux runcon**.

- Ver documentação sobre o comando original:

```
tldr -p linux runcon
```

# gsed

Este comando é um pseudônimo de **-p linux sed**.

- Ver documentação sobre o comando original:

```
tldr -p linux sed
```



# gseq

Este comando é um pseudônimo de **-p linux seq**.

- Ver documentação sobre o comando original:

```
tldr -p linux seq
```

# gsha1sum

Este comando é um pseudônimo de **-p linux sha1sum**.

- Ver documentação sobre o comando original:

```
tldr -p linux sha1sum
```

# gsha224sum

Este comando é um pseudônimo de **-p linux sha224sum**.

- Ver documentação sobre o comando original:

```
tldr -p linux sha224sum
```

# gsha256sum

Este comando é um pseudônimo de **-p linux sha256sum**.

- Ver documentação sobre o comando original:

```
tldr -p linux sha256sum
```

# gsha384sum

Este comando é um pseudônimo de **-p linux sha384sum**.

- Ver documentação sobre o comando original:

```
tldr -p linux sha384sum
```

# gsha512sum

Este comando é um pseudônimo de **-p linux sha512sum**.

- Ver documentação sobre o comando original:

```
tldr -p linux sha512sum
```

# gshred

Este comando é um pseudônimo de **-p linux shred**.

- Ver documentação sobre o comando original:

```
tldr -p linux shred
```

# gshuf

Este comando é um pseudônimo de **-p linux shuf**.

- Ver documentação sobre o comando original:

```
tldr -p linux shuf
```



# gsleep

Este comando é um pseudônimo de **-p linux sleep**.

- Ver documentação sobre o comando original:

```
tldr -p linux sleep
```

# gsort

Este comando é um pseudônimo de **-p linux sort**.

- Ver documentação sobre o comando original:

```
tldr -p linux sort
```

# gsplit

Este comando é um pseudônimo de **-p linux split**.

- Ver documentação sobre o comando original:

```
tldr -p linux split
```

# gstat

Este comando é um pseudônimo de **-p linux stat**.

- Ver documentação sobre o comando original:

```
tldr -p linux stat
```

# gstdbuf

Este comando é um pseudônimo de **-p linux stdbuf**.

- Ver documentação sobre o comando original:

```
tldr -p linux stdbuf
```

# gstty

Este comando é um pseudônimo de **-p linux stty**.

- Ver documentação sobre o comando original:

```
tldr -p linux stty
```

# gsum

Este comando é um pseudônimo de **-p linux sum**.

- Ver documentação sobre o comando original:

```
tldr -p linux sum
```

# gsync

Este comando é um pseudônimo de **-p linux sync**.

- Ver documentação sobre o comando original:

```
tldr -p linux sync
```



# gtac

Este comando é um pseudônimo de **-p linux tac**.

- Ver documentação sobre o comando original:

```
tldr -p linux tac
```

# gtail

Este comando é um pseudônimo de **-p linux tail**.

- Ver documentação sobre o comando original:

```
tldr -p linux tail
```

# gtalk

Este comando é um pseudônimo de **-p linux talk**.

- Ver documentação sobre o comando original:

```
tldr -p linux talk
```

# gtar

Este comando é um pseudônimo de **-p linux tar**.

- Ver documentação sobre o comando original:

```
tldr -p linux tar
```

# gtee

Este comando é um pseudônimo de **-p linux tee**.

- Ver documentação sobre o comando original:

```
tldr -p linux tee
```

# gtelnet

Este comando é um pseudônimo de **-p linux telnet**.

- Ver documentação sobre o comando original:

```
tldr -p linux telnet
```

# gtest

Este comando é um pseudônimo de **-p linux test**.

- Ver documentação sobre o comando original:

```
tldr -p linux test
```

# gtftp

Este comando é um pseudônimo de **-p linux tftp**.

- Ver documentação sobre o comando original:

```
tldr -p linux tftp
```



# gtime

Este comando é um pseudônimo de **-p linux time**.

- Ver documentação sobre o comando original:

```
tldr -p linux time
```

# gtimeout

Este comando é um pseudônimo de **-p linux timeout**.

- Ver documentação sobre o comando original:

```
tldr -p linux timeout
```

# gtouch

Este comando é um pseudônimo de **-p linux touch**.

- Ver documentação sobre o comando original:

```
tldr -p linux touch
```

# gtr

Este comando é um pseudônimo de **-p linux tr**.

- Ver documentação sobre o comando original:

```
tldr -p linux tr
```

# gtracroute

Este comando é um pseudônimo de **-p linux traceroute**.

- Ver documentação sobre o comando original:

```
tldr -p linux traceroute
```

# gtrue

Este comando é um pseudônimo de **-p linux true**.

- Ver documentação sobre o comando original:

```
tldr -p linux true
```

# gtruncate

Este comando é um pseudônimo de **-p linux truncate**.

- Ver documentação sobre o comando original:

```
tldr -p linux truncate
```

# gtsort

Este comando é um pseudônimo de **-p linux tsort**.

- Ver documentação sobre o comando original:

```
tldr -p linux tsort
```



# gtty

Este comando é um pseudônimo de **-p linux tty**.

- Ver documentação sobre o comando original:

```
tldr -p linux tty
```

# guname

Este comando é um pseudônimo de **-p linux uname**.

- Ver documentação sobre o comando original:

```
tldr -p linux uname
```

# gunexpand

Este comando é um pseudônimo de **-p linux unexpand**.

- Ver documentação sobre o comando original:

```
tldr -p linux unexpand
```

# guniq

Este comando é um pseudônimo de **-p linux uniq**.

- Ver documentação sobre o comando original:

```
tldr -p linux uniq
```

# gunits

Este comando é um pseudônimo de **-p linux units**.

- Ver documentação sobre o comando original:

```
tldr -p linux units
```

# gunlink

Este comando é um pseudônimo de **-p linux unlink**.

- Ver documentação sobre o comando original:

```
tldr -p linux unlink
```

# gupdatedb

Este comando é um pseudônimo de **-p linux updatedb**.

- Ver documentação sobre o comando original:

```
tldr -p linux updatedb
```

# guptime

Este comando é um pseudônimo de **-p linux uptime**.

- Ver documentação sobre o comando original:

```
tldr -p linux uptime
```



# gusers

Este comando é um pseudônimo de **-p linux users**.

- Ver documentação sobre o comando original:

```
tldr -p linux users
```

# gvdir

Este comando é um pseudônimo de **-p linux vdir**.

- Ver documentação sobre o comando original:

```
tldr -p linux vdir
```

# gwc

Este comando é um pseudônimo de **-p linux wc**.

- Ver documentação sobre o comando original:

```
tldr -p linux wc
```

# gwhich

Este comando é um pseudônimo de **-p linux which**.

- Ver documentação sobre o comando original:

```
tldr -p linux which
```

# gwho

Este comando é um pseudônimo de **-p linux who**.

- Ver documentação sobre o comando original:

```
tldr -p linux who
```

# gwhoami

Este comando é um pseudônimo de **-p linux whoami**.

- Ver documentação sobre o comando original:

```
tldr -p linux whoami
```

# gwhois

Este comando é um pseudônimo de **-p linux whois**.

- Ver documentação sobre o comando original:

```
tldr -p linux whois
```

# gxargs

Este comando é um pseudônimo de **-p linux xargs**.

- Ver documentação sobre o comando original:

```
tldr -p linux xargs
```



# gyes

Este comando é um pseudônimo de **-p linux yes**.

- Ver documentação sobre o comando original:

```
tldr -p linux yes
```

# hdiutil

Utilitário para criar e gerenciar imagens de disco.

Mais informações: <https://ss64.com/osx/hdiutil.html>.

- Monta uma imagem:

```
hdiutil attach {{caminho/para/arquivo_de_imagem}}
```

- Desmonta uma imagem:

```
hdiutil detach /Volumes/{{nome_do_volume}}
```

- Lista as imagens montadas:

```
hdiutil info
```

- Cria uma imagem ISO a partir do conteúdo de um diretório:

```
hdiutil makehybrid -o {{caminho/para/arquivo_de_saída}}  
{{caminho/para/diretório}}
```

# hidd

Daemon de nível de usuário da biblioteca HID.

Não deve ser invocado manualmente.

Mais informações: <https://www.manpagez.com/man/8/hidd/>.

- Inicia o daemon:

```
hidd
```

# icalBuddy

Utilitário de linha de comando para exibir eventos e tarefas do banco de dados do calendário do macOS.

Mais informações: <https://hasseg.org/icalBuddy/>.

- Exibe eventos que acontecerão hoje:

```
icalBuddy --includeOnlyEventsFromNowOn eventsToday
```

- Exibe tarefas incompletas:

```
icalBuddy uncompletedTasks
```

- Exibe uma lista formatada separada por calendário para todos os eventos de hoje:

```
icalBuddy --formatOutput --separateByCalendar eventsToday
```

- Exibe tarefas para um determinado número de dias:

```
icalBuddy --includeOnlyEventsFromNowOn "tasksDueBefore:today+{{dias}}"
```

- Exibe eventos em um intervalo de tempo:

```
icalBuddy eventsFrom:{{data_inicial}} to:{{data_final}}
```

# indent

Altera a aparência de um programa C/C++ inserindo ou excluindo espaços em branco.

Mais informações: <https://www.freebsd.org/cgi/man.cgi?query=indent>.

- Formata código fonte C/C++ de acordo com o estilo Berkeley:

```
indent {{caminho/para/fonte.c}} {{caminho/para/
fonte_identado.c}} -nbad -nbap -bc -br -c33 -cd33 -cdb -ce -
ci4 -cli0 -di16 -fc1 -fcb -i4 -ip -l75 -lp -npcs -nprs -psl -
sc -nsob -ts8
```

- Formata código fonte C/C++ de acordo com o estilo Kernighan & Ritchie (K&R):

```
indent {{caminho/para/fonte.c}} {{caminho/para/
fonte_identado.c}} -nbad -bap -nbc -br -c33 -cd33 -ncdb -ce -
ci4 -cli0 -cs -d0 -di1 -nfc1 -nfcb -i4 -nip -l75 -lp -npcs -
nprs -npsl -nsc -nsob
```

# InternetSharing

Configura o Compartilhamento de Internet.

Não deve ser invocado manualmente.

Mais informações: <https://www.manpagez.com/man/8/InternetSharing/>.

- Inicia o daemon:

`InternetSharing`

# istats

Ferramenta CLI que mostra estatísticas como temperatura da CPU, velocidade das ventoinhas, e status da bateria.

Mais informações: <https://github.com/Chris911/iStats>.

- Exibe todas as estatísticas:

```
istats
```

- Exibe todas as estatísticas da CPU:

```
istats cpu
```

- Exibe todas as estatísticas das ventoinhas:

```
istats fan
```

- Examina e imprime as temperaturas:

```
istats scan
```

# launchd

Este comando é um pseudônimo de **launchctl**.

Mais informações: <https://developer.apple.com/library/archive/documentation/MacOSX/Conceptual/BPSystemStartup/Chapters/Introduction.html>.

- Ver documentação sobre o comando original:

```
tldr launchctl
```



# xcodes runtimes

Gerencia runtimes do Simulador Xcode.

Mais informações: <https://github.com/xcodesorg/xcodes>.

- Lista todos os runtimes do Simulador disponíveis:

```
xcodes runtimes --include-betas
```

- Baixa um runtime do Simulador:

```
xcodes runtimes download {{nome-do-runtime}}
```

- Baixa e instala um runtime do Simulador:

```
xcodes runtimes install {{nome-do-runtime}}
```

# xcodes

Baixe, instale e gerencie várias versões do Xcode.

Veja também: **xcodes** **runtimes**.

Mais informações: <https://github.com/xcodesorg/xcodes>.

- Lista todas as versões do Xcode instaladas:

```
xcodes installed
```

- Lista todas as versões do Xcode disponíveis:

```
xcodes list
```

- Seleciona uma versão do Xcode especificando o número da versão ou um caminho:

```
xcodes select {{versao-do-xcode|caminho/para/Xcode.app}}
```

- Baixa e instala uma versão específica do Xcode:

```
xcodes install {{versao-do-xcode}}
```

- Baixa, instala e seleciona a versão mais recente do Xcode:

```
xcodes install --latest --select
```

- Baixa uma versão específica do Xcode para um diretório específico sem instalá-la:

```
xcodes download {{versao-do-xcode}} --directory {{caminho/para/diretorio}}
```

Windows

# assoc

Exibir ou alterar associações entre extensões de arquivo e tipos de arquivos.

Mais informações: <https://learn.microsoft.com/windows-server/administration/windows-commands/assoc>.

- Lista todas as associações entre extensões de arquivo e tipos de arquivos:

```
assoc
```

- Exibe o tipo de arquivo associado para uma extensão específica:

```
assoc {{.txt}}
```

- Define o tipo de arquivo associado para uma extensão específica:

```
assoc .{{txt}}={{arquivotxt}}
```

# cd

Exibe o nome ou altera o diretório local atual.

Mais informações: <https://learn.microsoft.com/windows-server/administration/windows-commands/cd>.

- Vá para um diretório na mesma unidade:

```
cd {{caminho/para/diretorio}}
```

- Mostra o nome do diretório atual:

```
cd
```

- Vá até o pai do diretório atual:

```
cd ..
```

- Vá para um diretório em uma unidade diferente:

```
cd {{caminho/para/diretorio}} /d
```

# choco info

Exibir informações detalhadas de um pacote com Chocolatey.

Mais informações: <https://chocolatey.org/docs/commands-info>.

- Exibir informações sobre um pacote específico:

```
choco info {{pacote}}
```

- Exibir informação para um pacote local:

```
choco info {{pacote}} --local-only
```

- Especificar uma fonte personalizada para receber as informações de um pacote:

```
choco info {{pacote}} --source {{url_da_fonte|apelido}}
```

- Fornecer um nome e uma senha para autenticação:

```
choco info {{pacote}} --user {{apelido}} --password {{senha}}
```

# choco install

Instalar um pacote ou mais com Chocolatey.

Mais informações: <https://chocolatey.org/docs/commands-install>.

- Instalar um ou mais pacotes separado por espaço:

```
choco install {{pacote(s)}}
```

- Instalar pacotes a partir de um arquivo de configuração personalizado:

```
choco install {{caminho/para/pacotes.config}}
```

- Instalar um arquivo específico nuspec ou nupkg:

```
choco install {{caminho/para/arquivo}}
```

- Instalar uma versão específica de um pacote:

```
choco install {{pacote}} --version {{versão}}
```

- Permitir a instalação de múltiplas versões de um pacote:

```
choco install {{pacote}} --allow-multiple
```

- Confirmar todos prompts automaticamente:

```
choco install {{pacote}} --yes
```

- Especificar uma fonte personalizada para receber pacotes:

```
choco install {{pacote}} --source {{url_do_pacote|apelido}}
```

- Fornecer um nome e uma senha para autenticação:

```
choco install {{pacote}} --user {{usuario}} --password {{senha}}
```

# choco list

Exibir uma lista de pacotes com Chocolatey.

Mais informações: <https://chocolatey.org/docs/commands-list>.

- Exibir todos pacotes disponíveis:

```
choco list
```

- Exibir todos pacotes instalados localmente:

```
choco list --local-only
```

- Exibir uma lista incluindo programas locais:

```
choco list --include-programs
```

- Exibir apenas pacotes aprovados:

```
choco list --approved-only
```

- Especificar uma fonte personalizada para exibir os pacotes:

```
choco list --source {{url_da_fonte|apelido}}
```

- Fornecer um nome e uma senha para autenticação:

```
choco list --user {{usuário}} --password {{senha}}
```



# choco source

Gerenciar fontes para pacotes com Chocolatey.

Mais informações: <https://chocolatey.org/docs/commands-source>.

- Listar fontes atualmente disponíveis:

```
choco source list
```

- Adicionar uma nova fonte de pacotes:

```
choco source add --name {{nome}} --source {{url_da_fonte}}
```

- Adicionar uma nova fonte de pacotes com credenciais:

```
choco source add --name {{nome}} --source {{url_da_fonte}} --  
user {{nome}} --password {{senha}}
```

- Adicionar uma nova fonte de pacotes com certificado do cliente:

```
choco source add --name {{nome}} --source {{url_da_fonte}} --  
cert {{caminho/para/certificado}}
```

- Habilitar uma fonte de pacotes:

```
choco source enable --name {{nome}}
```

- Desabilitar uma fonte de pacotes:

```
choco source disable --name {{nome}}
```

- Remover uma fonte de pacotes:

```
choco source remove --name {{nome}}
```

# choco uninstall

Desinstalar um pacote ou mais com Chocolatey.

Mais informações: <https://chocolatey.org/docs/commands-uninstall>.

- Desinstalar um pacote ou mais separado por espaços:

```
choco uninstall {{pacote(s)}}
```

- Desinstalar uma versão específica de um pacote:

```
choco uninstall {{pacote}} --version {{versão}}
```

- Confirmar todos prompts automaticamente:

```
choco uninstall {{pacote}} --yes
```

- Remover todas dependências ao desinstalar:

```
choco uninstall {{pacote}} --remove-dependencies
```

- Desinstalar todos os pacotes:

```
choco uninstall all
```

# choco upgrade

Atualizar um ou mais pacotes com Chocolatey.

Mais informações: <https://chocolatey.org/docs/commands-upgrade>.

- Atualizar um ou mais pacotes separados por espaço:

```
choco upgrade {{pacote(s)}}
```

- Atualizar para uma versão específica de um pacote:

```
choco upgrade {{pacote}} --version {{versão}}
```

- Atualizar todos pacotes:

```
choco upgrade all
```

- Atualizar todos os pacotes, exceto os especificados separados por virgula:

```
choco upgrade all --except "{{pacote(s)}}"
```

- Confirmar todos prompts automaticamente:

```
choco upgrade {{pacote}} --yes
```

- Especifique uma fonte personalizada para receber pacotes:

```
choco upgrade {{pacote}} --source {{url_do_pacote|apelido}}
```

- Fornecer um nome e uma senha para autenticação:

```
choco upgrade {{pacote}} --user {{usuário}} --password {{senha}}
```

# choco

Uma interface de linha de comando para o gerenciador de pacotes Chocolatey.

Alguns subcomandos como **choco install** tem sua própria documentação de uso.

Mais informações: <https://chocolatey.org>.

- Executar o comando Chocolatey:

```
choco {{comando}}
```

- Exibir ajuda generalizada:

```
choco -?
```

- Exibir ajuda em um comando específico:

```
choco {{comando}} -?
```

- Exibir a versão do Chocolatey:

```
choco --version
```

# chrome

Este comando é um pseudônimo de **chromium**.

Mais informações: <https://chrome.google.com>.

- Ver documentação sobre o comando original:

```
tldr chromium
```

# cinst

Este comando é um pseudônimo de **choco install**.

Mais informações: <https://docs.chocolatey.org/en-us/choco/commands/install>.

- Ver documentação sobre o comando original:

```
tldr choco install
```

# clist

Este comando é um pseudônimo de **choco list**.

Mais informações: <https://docs.chocolatey.org/en-us/choco/commands/list>.

- Ver documentação sobre o comando original:

```
tldr choco list
```

# cls

Limpar a tela de saída.

Mais informações: <https://learn.microsoft.com/windows-server/administration/windows-commands/cls>.

- Limpar a tela:

```
cls
```



# cmd

O interpretador de comandos do Windows.

Mais informações: <https://learn.microsoft.com/windows-server/administration/windows-commands/cmd>.

- Iniciar nova instância do interpretador de comandos:

```
cmd
```

- Executar o comando especificado e sair do interpretador:

```
cmd /c "{{comando}}"
```

- Executar o comando especificado e entrar no shell interativo:

```
cmd /k "{{comando}}"
```

- Desabilitar o uso do comando `echo` na saída dos comandos:

```
cmd /q
```

- Habilitar ou desabilitar extensão de comandos:

```
cmd /e:{{on|off}}
```

- Habilitar ou desabilitar a ferramenta que completa automaticamente o nome de arquivos ou diretórios:

```
cmd /f:{{on|off}}
```

- Habilitar ou desabilitar a expansão de variáveis de ambiente:

```
cmd /v:{{on|off}}
```

- Forçar que a saída de comandos use o padrão Unicode:

```
cmd /u
```

# cpush

Este comando é um pseudônimo de **choco-push**.

Mais informações: <https://docs.chocolatey.org/en-us/create/commands/push>.

- Ver documentação sobre o comando original:

```
tldr choco-push
```

# cuninst

Este comando é um pseudônimo de **choco uninstall**.

Mais informações: <https://docs.chocolatey.org/en-us/choco/commands/uninstall>.

- Ver documentação sobre o comando original:

```
tldr choco uninstall
```

# curl

Este comando é um pseudônimo de `curl -p common`.

Mais informações: <https://learn.microsoft.com/powershell/module/microsoft.powershell.utility/invoke-webrequest>.

- Ver documentação sobre o comando original:

```
tldr curl -p common
```

# dir

Listar os conteúdos de um diretório.

Mais informações: <https://learn.microsoft.com/windows-server/administration/windows-commands/dir>.

- Mostrar o conteúdo do diretório atual:

```
dir
```

- Mostrar o conteúdo do diretório no caminho provido pelo usuário:

```
dir {{caminho/para/diretório}}
```

- Mostrar o conteúdo do diretório atual, incluindo arquivos e pastas escondidas:

```
dir /A
```

- Mostrar o conteúdo do diretório provido pelo usuário, incluindo arquivos e pastas escondidas:

```
dir {{caminho/para/diretório}} /A
```

# iwr

Este comando é um pseudônimo de **invoke-webrequest**.

Mais informações: <https://learn.microsoft.com/powershell/module/microsoft.powershell.utility/invoke-webrequest>.

- Ver documentação sobre o comando original:

```
tldr invoke-webrequest
```

# mkdir

Criar um diretório.

Mais informações: <https://learn.microsoft.com/windows-server/administration/windows-commands/mkdir>.

- Criar um diretório:

```
mkdir {{nome_do_diretorio}}
```

- Criar recursivamente uma árvore de diretórios aninhados:

```
mkdir {{caminho/para/subdiretorio}}
```

# print

Imprimir um arquivo de texto em uma impressora.

Mais informações: <https://learn.microsoft.com/windows-server/administration/windows-commands/print>.

- Imprimir um arquivo de texto na impressora padrão:

```
print {{caminho/para/arquivo}}
```

- Imprimir arquivo de texto em uma impressora específica:

```
print /d:{{impressora}} {{caminho/para/arquivo}}
```



# pwsh-where

Este comando é um pseudônimo de **Where-Object**.

Mais informações: <https://learn.microsoft.com/powershell/module/microsoft.powershell.core/where-object>.

- Ver documentação sobre o comando original:

`tldr Where-Object`

# rd

Este comando é um pseudônimo de **rmdir**.

Mais informações: <https://learn.microsoft.com/windows-server/administration/windows-commands/rd>.

- Ver documentação sobre o comando original:

```
tldr rmdir
```

# sls

Este comando é um pseudônimo de **Select-String**.

Mais informações: <https://learn.microsoft.com/powershell/module/microsoft.powershell.utility/select-string>.

- Ver documentação sobre o comando original:

```
tldr select-string
```

# title

Exibe o título do prompt de comando.

Mais informações: <https://learn.microsoft.com/windows-server/administration/windows-commands/title>.

- Define o título do prompt de comando:

```
title {{novo_título}}
```

# tree

Exibe uma árvore gráfica da estrutura do diretório no caminho.

Mais informações: <https://learn.microsoft.com/windows-server/administration/windows-commands/tree>.

- Exibe a árvore para o diretório atual:

```
tree
```

- Exibe a árvore para o diretório específico:

```
tree {{caminho/para/diretório}}
```

- Exibe a árvore para o diretório específico incluindo arquivos:

```
tree {{caminho/para/diretório}} /f
```

- Exibe a árvore usando caracteres ASCII:

```
tree {{caminho/para/diretório}} /a
```

# type

Mostrar o conteúdo de um arquivo.

Mais informações: <https://learn.microsoft.com/windows-server/administration/windows-commands/type>.

- Mostrar o conteúdo de um arquivo específico:

```
type {{caminho/para/arquivo}}
```

# ver

Exibe a atual versão do Windows ou MS-DOS.

Mais informações: <https://learn.microsoft.com/windows-server/administration/windows-commands/ver>.

- Mostra a atual versão:

`ver`

# wget

Este comando é um pseudônimo de `wget -p common`.

Mais informações: <https://learn.microsoft.com/powershell/module/microsoft.powershell.utility/invoke-webrequest>.

- Ver documentação sobre o comando original:

```
tldr wget -p common
```



# whoami

Mostra detalhes sobre o usuário atual.

Mais informações: <https://learn.microsoft.com/windows-server/administration/windows-commands/whoami>.

- Mostra o username do usuário atual:

```
whoami
```

- Mostra os grupos dos quais o usuário atual faz parte:

```
whoami /groups
```

- Mostra os privilégios do usuário atual:

```
whoami /priv
```

- Mostra o nome principal (UPN) do usuário atual:

```
whoami /upn
```

- Mostra o id de logon do usuário atual:

```
whoami /logonid
```

