

Mac

Cositas de macos y apple

- [Montar un punto de montaje NFS en MacOS 12 Monterey](#)
- [Configurar un servidor APF con netatalk en una Raspberry Pi](#)
- [Cómo acceder a una máquina Linux con RDP en un Mac](#)
- [chatr con Apple MacOS X](#)
- [Extraer páginas de un PDF en MacOS X desde el shell](#)
- [Secuencia de Apple Script para cambiar colores en Numbers](#)
- [Cómo agregar tu firma de correo electrónico HTML en la aplicación Mac Mail.app](#)
- [Force Sleep en macOS — LaunchAgent con pmset sleepnow](#)
- [Tips rapidos.](#)

Montar un punto de montaje NFS en MacOs 12 Monterey

Introducción

A veces se me olvida que Mac tiene sus cositas. En mi caso tengo una raspberry pi 4B que es mi centro de backups, DHCP, etc. Habitualmente monto alguna unidad vía NFS con mi ordenador personal con linux, y como no quería hacer lo propio con mi Macbook Pro con chip M1 y Monterey (12.5) como versión del SO

“ Tras unas horas de trabajo, he comprendido que es mucho mejor instalar un [servidor APF con Netatalk](#) y permitir el uso de su formato nativo de red, que usar NFS que se le atraganta mucho al Darwin.

Metodología

Para ello lo primero me cree los directorios que queria (uno por cada endpoint NFS) y luego intente montar uno

```
> sudo mkdir /private/backup5t
> sudo make mount -t nfs rbp:/backup5t /private/backup5t
Password:
mount_nfs: can't mount with remote locks when server (rbp) is not running rpc.statd: RPC prog.
not avail
mount: /private/backup5t failed with 74
```

Bueno, algo me decía que empezábamos un nuevo lío.

Verificar NFS remoto

```
> showmount -e rbp
Exports list on rbp:
/backup5t                192.168.1.0/24
```

Verificar los demonios asociados a nfs

```
> systemctl status rpcbind.service
● rpcbind.service - RPC bind portmap service
   Loaded: loaded (/lib/systemd/system/rpcbind.service; enabled; vendor preset: enabled)
   Active: active (running) since Sun 2022-08-07 08:26:48 UTC; 42min ago
     Docs: man:rpcbind(8)
 Main PID: 279 (rpcbind)
    Tasks: 1 (limit: 4915)
   Memory: 2.1M
   CGroup: /system.slice/rpcbind.service
           └─279 /sbin/rpcbind -f -w
> systemctl status rpc-statd.service
● rpc-statd.service - NFS status monitor for NFSv2/3 locking.
   Loaded: loaded (/lib/systemd/system/rpc-statd.service; disabled; vendor preset: enabled)
   Active: inactive (dead)
```

Aquí lo vemos. Un servicio de locking que está muerto pero es normal, ya que no tengo configurado nfs v3 o v2.

Y por aquí es donde anda el lio.

Solución

Decirle a nuestro mác que use la versión 4 o configurar el servidor para que soporte nfs v3 y nfs4, que no estoy dispuesto, por obsoleto.

```
> sudo echo 'nfs.client.mount.options = vers=4' >> /etc/nfs.conf
> sudo mount -t nfs rbp:/backup5t /private/backup5t
```

Et voila...

Disco NFS montado en MacOS X Monterey

Solventar el lío a lo bruto

La literatura sobre el tema en Google, o DuckDuck es fantástica. Una vez más muchos posts, y pocas soluciones reales, amén de que muchas son **similares** pero no tienen la misma dinámica. Y esta es una de las propuestas, que tiene un gran handicap.

Su uso es desaconsejado porque en un entorno de múltiples accesos a un mismo fichero acabarán en una corrupción NO USAR NUNCA nolocks

```
> sudo mount -o nolocks,resvport,locallocks -t nfs rbp:/backup5t /private/backup5t
```

Aviso

Esta documentación y su contenido, no implica que funcione en tu caso o determinados casos. También implica que tienes conocimientos sobre lo que trata, y que en cualquier caso tienes copias de seguridad. El contenido el contenido se entrega, tal y como está, sin que ello implique ningún obligación ni responsabilidad por parte de [Castris](#)

Si necesitas soporte profesional puedes contratar con Castris [soporte profesional](#).

Configurar un servidor APF con netatalk en una Raspberry Pi

Introducción

Tras un rato con la configuración del **NFS v4** para conectarme desde mi Macbook Pro a mi raspberrypi (linux con [Hyprion](#), he visto que al final perderé el tiempo, la velocidad no es muy agraciada, y encima se me olvidó un detalle, necesito una partición en un de mis discos atados a la raspberry que tengo formato **hfsplus** o **apfs**. A este punto no encuentre nada sobre APFS para linux excepto un repositorio de github [Apple File System for Linux](#) y yo soy friki pero no idiota como para perder mi tiempo, y menos cuando una de las particiones va a ser mi **Time Machine** o backup.

Metodología

Debemos realizar los siguientes pasos: Instalar el soporte de **HPFS+** Instalar soporte **netatalk** para poder comunicarnos con el protocolo de red de Apple. Particionar el disco con **parted** ya que el disco de **Time Machine** debe estar formateado y montado como una unidad HPFS+ Modificar el fichero de montaje fstab para que sea permanente Editar la configuración de netatalk

Después ya podremos comprobar el trabajo realizado.

En el ejemplo vamos a usar un disco de 5TB con dos particiones, una para **hfs+** y **xf**

Soporte HPFS+

Actualización del sistema de paquetes

```
sudo apt-get update
sudo apt-get upgrade
```

Instalación de paquetes para HPFS+

Sin ellos no podremos dar soporte a nuestro linux para que lea y escriba en un sistema de ficheros HPFS+

```
sudo apt-get install hfsplus hfsutils hfsprogs gdisk
```

Crear una particion HPFS+

Primero necesitamos crear una partición para usar con dicho sistema de ficheros.

En mi caso voy a formatear un disco duro de 5TB en dos particiones. Una para **xf**s y otra para **hpfs+*

Es un disco que actualmente tenía dos particiones, una reservada que no borre y otra xfs. En el proceso voy a eliminar las dos particiones para luego crear dos particiones conforme a mis deseos, usando **parted**

Más información [Parted mejor que fdisk](#)

```
> sudo parted /dev/sdb
GNU Parted 3.2
Using /dev/sdb
Welcome to GNU Parted! Type 'help' to view a list of commands.
(parted) print
Model: Seagate Expansion Desk (scsi)
Disk /dev/sdb: 6001GB
Sector size (logical/physical): 512B/4096B
Partition Table: gpt
Disk Flags:

Number  Start   End     Size    File system  Name              Flags
  1      17.4kB  134MB   134MB                   Microsoft reserved partition  msftres
  2      135MB  6001GB  6001GB   xfs          BACKUP5T

(parted) rm 2
(parted) rm 1
(parted) print
Model: Seagate Expansion Desk (scsi)
Disk /dev/sdb: 6001GB
Sector size (logical/physical): 512B/4096B
Partition Table: gpt
Disk Flags:
```

```
Number Start End Size File system Name Flags
```

```
(parted) quit
```

```
Information: You may need to update /etc/fstab.
```

```
> sudo parted -s /dev/sdb mkpart backup3t 0% 50%
```

```
> sudo parted -s /dev/sdb mkpart timemachine 50% 100%
```

```
> sudo parted /dev/sdb
```

```
GNU Parted 3.2
```

```
Using /dev/sdb
```

```
Welcome to GNU Parted! Type 'help' to view a list of commands.
```

```
(parted) print
```

```
Model: Seagate Expansion Desk (scsi)
```

```
Disk /dev/sdb: 6001GB
```

```
Sector size (logical/physical): 512B/4096B
```

```
Partition Table: gpt
```

```
Disk Flags:
```

Number	Start	End	Size	File system	Name	Flags
2	1049kB	3001GB	3001GB		backup3t	
1	3001GB	6001GB	3001GB		timemachine	

```
(parted) quit
```

Que bien que ahora podemos formatear las dos particiones.

```
> sudo mkfs.hfsplus /dev/sdb2
```

```
Initialized /dev/sdb2 as a 2795 GB HFS Plus volume
```

```
> sudo mkfs.xfs /dev/sdb1
```

```
meta-data=/dev/sdb1          isize=512    agcount=4, agsize=183141312 blks
      =                       sectsz=4096  attr=2, projid32bit=1
      =                       crc=1          finobt=1, sparse=1, rmapbt=0
      =                       reflink=0
data      =                   bsize=4096  blocks=732565248, imaxpct=5
      =                       sunit=0        swidth=0 blks
naming    =version 2          bsize=4096  ascii-ci=0, ftype=1
```

```
log      =internal log          bsize=4096  blocks=357697, version=2
      =                          sectsz=4096  sunit=1 blks, lazy-count=1
realtime =none                  extsz=4096  blocks=0, rtextents=0
```

Crear las particiones de montaje

Creamos las particiones de montaje para ambos sistemas, y ya que van a ser usadas en un entorno personal le damos los permisos adecuados

```
> sudo mkdir /timemachine
> sudo mkdir /backup3t
```

Probar el montaje manual

Primero probamos manualmente el montaje, y luego pasaremos a el montaje automático en cada reinicio.

“ Aconsejado cuando hacemos cosas con nuevos dispositivos, y sistemas de ficheros, para liberarnos de problemas en el montaje automático.

El comando de abajo, formateara la partición con las siguientes características (disponible la ayuda `man de man mkfs.hfsplus`)

- **s** Que creará un formato sensitivo al caso (distinguir mayúsculas de minúsculas)
- **v** nombre del volumen

“ Lo de Apple y las mayusculas y minusculas es que no me entra. Un sistema UNIX como es MacOSX basado en Darwin, en el que coexiste con otros UNIX, que por defecto te instala un sistema sin distinción de caso.

“ La opción **J** Crearía el volumen con un fichero de diario o journal en este caso de 512M (máximo). Yo lo considero necesario pues en caso de un problema con el disco en un reinicio, corte de electricidad, es una garantía muy fuerte de que se recuperara la información, y en backups basados en rsync como los es el Time Machine, es más que necesario por la cantidad tan gigante de información que

se genera en el journal. Lamentablemente a la fecha no conozco soporte para instalar el journal ya que al montaje, el disco se haria indisponible por un problema de permisos sobre el journal.

```
> sudo mkfs.hfsplus -s -v timemachine /dev/sdb1
Initialized /dev/sdb1 as a 2795 GB HFS Plus volume

> sudo fsck.hfsplus /dev/sdb1
** /dev/sdb1
** Checking HFS Plus volume.
** Detected a case-sensitive catalog.
** Checking Extents Overflow file.
** Checking Catalog file.
** Checking Catalog hierarchy.
** Checking volume bitmap.
** Checking volume information.
** The volume timemachine appears to be OK.
```

Montaje hfsplus manual

Montaje automatizado de los discos

Una vez montado podemos ver las opciones del montaje que estarán disponibles en el `/etc/mtab`

```
/dev/sdb1 /timemachine hfsplus rw,relatime,umask=22,uid=0,gid=0,nls=utf8 0 0
```

Editamos el fichero `/etc/fstab`

```
> sudo cat /etc/fstab |grep time
/dev/sdb1 /timemachine hfsplus rw,relatime,umask=22,uid=0,gid=0,nls=utf8 0 1
```

Con ellas ya tenemos una idea de que poner en el `/etc/fstab`

Obtención del UUID

Suelo montar las particiones con UUID y con más razón cuando son discos externos, ya que un cambio, eliminación de dispositivo, error, en un reinicio puede cambiar las letras del dispositivo.

```
> sudo blkid
```

Comando blkid

Edición del fstab

```
UUID=6bc6f79f-c74f-35a1-b226-092396e3fed1 /timemachine hfsplus
rw,relatime,umask=22,uid=0,gid=0,nls=utf8 0 1
UUID=c0f66c48-3a78-4c1a-bc7f-6d4ad854f284 /backup3t xfs rw,noquota,nofail 0 1
```

Configuración de netatalk

Una buen guía de netatalk es la [Guía de Archlinux | Netatalk](#) aunque es bueno darse una leída al propio proyecto [Netatalk](#)

Abajo mi configuración

```
> cat /etc/netatalk/afp.conf
;
; Netatalk 3.x configuration file
;

[Global]
; Global server settings
; mimic model = TimeCapsule6,106
log level = default:warn
log file = /var/log/afpd.log
hosts allow = 192.168.1.0/24
spotlight = yes

; [Homes]
; basedir regex = /home

[Pirate]
path = /home/pirate
valid users = pirate

[Backup10t]
path = /backup10t
valid users = pirate
```

```
[Backup3t]
path = /backup3t
valid users = pirate
```

```
[My Time Machine Volume]
path = /timemachine
time machine = yes
```

Reiniciar **netatalk** con `sudo systemctl restart netatalk` ya tendremos disponibles nuestras particiones en el mac a través de las unidades de red.

“ Nota: El reinicio de netatalk tras algún cambio o el inicio por primera vez, tardará dependiendo de la cantidad de contenidos de las unidades compartidas configuradas.

Montaje de las particiones

Ya está, ya solo tenemos conectarnos al servidor y montar nuestras particiones.

Desde el Finder > Ir > Conectarse a un servidor (⌘)

Conectarse a un servidor APF Conectarse un volumen de un servidor APF

“ NOTA: La primera vez que se conecta a un servidor APF tardará un buen rato en terminar la conexión. Sé paciente, sobre todo si tus discos o particiones APF tiene un gran volumen de ficheros y enlaces, ya que este sistema de red, tiene una especie de journal, que tiene que completar. No te desesperes, en mi caso la primera vez tardó casi 10 minutos, ya que el disco `backup10t` tiene casi 8TB de backups con rsync de más de 2 años, lo que hace que su número de inodos usados en enlaces duros sea muy elevado.

Configurar Time Machine

Ahora ya podemos configurar nuestro Time Machine **Preferencias > Time Machine** y disfrutar delo aprendido.

Disclaimer o descargo de responsabilidad

Esto es un documento tipo “Yo lo hice así y me funciona”. Eso supone que cuando menos tienes conocimientos, gustas de leer la documentación y tratar de entenderla. Que eres un forofo de las copias de seguridad y que por supuesto, eres consciente de que si haces algo mal la puedes liar. No me vengas luego con llantos. Si te pasa algo yo no soy el culpable. ¿Entendido? Que los disfrutes.

Aviso

Esta documentación y su contenido, no implica que funcione en tu caso o determinados casos. También implica que tienes conocimientos sobre lo que trata, y que en cualquier caso tienes copias de seguridad. El contenido el contenido se entrega, tal y como está, sin que ello implique ningún obligación ni responsabilidad por parte de [Castris](#)

Si necesitas soporte profesional puedes contratar con Castris [soporte profesional](#).

Cómo acceder a una máquina Linux con RDP en un Mac

Introducción

Aunque he vuelto a la casa Apple con mi nuevo max, sigo teniendo un montón de máquinas y dispositivos basados en Linux, en mi oficina profesional. Tenía oxidado el RDP y como no, tras leer un par de artículos me encuentro lo de siempre: incompletos al menos para un escenario como el mio, con dos pantallas en el ordenador al que quiero conectar, mi antiguo ordenador de sobremesa, ahora en manos de otra persona, pero en el que aún tengo cosas que todavía no puedo trasladar operativamente a mi mac. (Cosas de Docker y otras hierbas)

Tras seguir los pasos de [How to RDP to Linux from a Mac computer](#) me encuentre con un desagradable logout inmediato tras el login.

“ Una cosa importante. A estas altura salvo que uno esté con un equipo antiguo, con pocos recursos, (cosa rara o cosa de una raspberry por ejemplo) lo de instalar el xfce, alterar tu ordenador y más si hay otros usuarios usandolo, es para poner un aviso grande en los blogs de expertos.

Instalación

Descargar el cliente RDP de la App Store. En mi caso, por conocerlo de hace ya 20 años, uso Microsoft. Gratuito. Instalar RDP en linux (Ubuntu 22.04 en este caso) Configurar la conexión en el cliente y si falla configurar RDP en linux

Instalar RDP

```
sudo apt update && sudo apt upgrade -y && sudo apt install xrdp && sudo systemctl enable xrdp
```

Conceder accesos a certificados

En la instalación tipo que nos indican los expertos.

```
[20220924-09:33:09] [INFO ] Using default X.509 certificate: /etc/xrdp/cert.pem
[20220924-09:33:09] [INFO ] Using default X.509 key file: /etc/xrdp/key.pem
[20220924-09:33:09] [ERROR] xrdp_iso_send: trans_write_copy_s failed
[20220924-09:33:09] [ERROR] Cannot read private key file /etc/xrdp/key.pem: Permission denied
```

Luego algo falla, y son permisos

```
sudo adduser xrdp ssl-cert
```

Ajustes en el fichero de configuración

```
cp /etc/xrdp/xrdp.ini /etc/xrdp/xrdp.ini.back
```

Editar

```
<editor> /etc/xrdp/xrdp.ini
```

```
# Uso en una red local segura, sin acceso remoto por internet. De lo contrario **high**
crypt_level=low
# Compresión
bitmap_cache=true
bitmap_compression=true
bulk_compression=true
max_bpp=32 # En mi caso local con fibra, me gusta trabajar cómodo, pero puedes reducir en
otros casos
```

Configurar el cliente Remote Desktop

Con las configuraciones de más abajo (para un pc con dos pantallas y un mac con una pantalla QHDC 3440 × 1440, me conectaron, pero me escupió del log en cuanto entre. .

Configuración RDP-Cliente :: General

Configuración RDP-Cliente :: Display

Disfrutar del Escritorio Remoto

RDP-Cliente :: Conexión Mac a Linux Ubuntu 22.04

Aviso

Esta documentación y su contenido, no implica que funcione en tu caso o determinados casos. También implica que tienes conocimientos sobre lo que trata, y que en cualquier caso tienes copias de seguridad. El contenido el contenido se entrega, tal y como está, sin que ello implique ningún obligación ni responsabilidad por parte de [Castris](#)

Si necesitas soporte profesional puedes contratar con Castris [soporte profesional](#).

chattr con Apple MacOS X

chattr en linux

Uno de los elementos que uso en mis sistemas linux, cuando hay ciertas cosas que quiero proteger de escrituras accidentales o provocadas, es `chattr` un poderoso comando que marca con una bandera de inmutabilidad el fichero indicado, si lo usamos con la opción `+i`

```
# chattr +i file_inmutable.txt
```

Con ese comando ni el propio root podrá editar o borrar el fichero hasta que

```
# chattr -i file_inmutable.txt
```

“ A file with the 'i' attribute cannot be modified: it cannot be deleted or renamed, no link can be created to this file and no data can be written to the file. Only the superuser or a process possessing the CAP_LINUX_IMMUTABLE capability can set or clear this attribute.

Sin embargo en mi oficina como ordenador de escritorio uso Mac OsX, y es un poco diferente. Te presento el comando `chflags`

chflags

Si acudimos al **man* del comando

```
schg, schange, simmutable
                set the system immutable flag (super-user only)
uchg, uchange, uimmutable
                set the user immutable flag (owner or super-user only)
```

Putting the letters “no” before or removing the letters “no” from a keyword causes the flag to be cleared. For example:

```
nouchg clear the user immutable flag (owner or super-user)
```

```
only)
```

Así que sencillo si queremos proteger contra una posible escritura o borrado un fichero en nuestro mac.

```
> chflags uchg config.xml
```

Aviso

Esta documentación y su contenido, no implica que funcione en tu caso o determinados casos. También implica que tienes conocimientos sobre lo que trata, y que en cualquier caso tienes copias de seguridad. El contenido el contenido se entrega, tal y como está, sin que ello implique ningún obligación ni responsabilidad por parte de [Castris](#)

Si necesitas soporte profesional puedes contratar con Castris [soporte profesional](#).

Extraer páginas de un PDF en MacOs X desde el shell

Antecedentes

Que si esta, que si la otra,... un baile de aplicaciones algunas de ellas muy engañosas con el concepto **gratis** cuando en realidad son aplicaciones capadas que buscan tu dinero.

Opensource que no es lo mismo y es mejor. En mi artículo [Extraer páginas de un PDF en Linux desde el shell](#) escribí sobre algo parecido para Linux. Hoy le toca el turno a Mac Os.

Tras un incidente con la falta de usabilidad de [Dominio.es](#) en el que te piden un DNI en el proceso de validación de un cambio de titular, tuve que extraer del PDF que me envió el cliente, las dos imágenes del DNI, y después, juntarlas en una sola imagen (no admiten más de una), para terminar con un, ****se me olvide decirte que la imagen no puede tener más de 600px de ancho)** realice el trabajo en el shell

Programa Poppler

[Poppler](#) es un programa de rendering de PDF basado en [xpdf](#)

Usaremos [Homebrew](#) para instalarlo.

```
brew install poppler
```

Con Poppler, extraeremos las imágenes a través de su binario **pdftimages**

```
pdftimages archivo.pdf prefijo_salida
```

- `archivo.pdf` es el nombre de tu archivo
- `prefijo_salida` es el prefijo que tendrán tus imágenes extraídas

Si lo hacemos así las imágenes saldrán en formato [.ppm](#) así que tendríamos que convertirlos a JPG pro imperativo de dominios.es.

Programa ImageMagick

Usaremos uno de los programas de ImageMagick, así que si no lo tenemos (es un super programa, tanto en Linux como en Mac) lo instalaremos.

```
brew install imagemagick
```

Una vez instalado de todos los binarios o programas que instala usaremos varios.

`convert` para convertirlo a otro formato

```
convert prefijo_salida-0000.ppm prefijo_salida-0000.jpg
```

Como tenemos mas de una imagen que bueno un comando

```
for img in *.ppm; do convert "$img" "${img%.ppm}.jpg"; done
```

Tip

Si antes de hacer nada leemos el man de **poppler** directamente podemos hacer la extracción a jpg.

Usamos el parametro `-j`

```
pdftimages -j archivo.pdf prefijo_salida
```

Crear un único archivo

Bueno, cuando vas a subir el fichero descubres que sólo quieren un fichero.

Nos toca usar `convert` otra vez para añadirlos.

```
convert prefijo_salida-0000.jpg prefijo_salida-0001.jpg +append resultado_final.jpg
```

Ay... que ahora no lo quieren de max de 600px de ancho.

```
convert prefijo_salida-0000.jpg prefijo_salida-0001.jpg -append resultado_final.jpg
```

Observa que `+append` se convirtió en `-append` para crearlo en vertical uno tras otro.

Pero, no. No comprobé que el ancho de cada imagen fuera de menos de 600px, y resultado que tenía 610.

Bueno, ya es el último.

```
convert resultado_final.jpg -resize 600x resultado_final_redimensionado.jpg
```

Finales

Espero que lo hallas disfrutado. Ahora te dejo que tengo que seguir con mi proyecto, [Baytuka](#)

Aviso

Esta documentación y su contenido, no implica que funcione en tu caso o determinados casos. También implica que tienes conocimientos sobre lo que trata, y que en cualquier caso tienes copias de seguridad. El contenido el contenido se entrega, tal y como está, sin que ello implique ningún obligación ni responsabilidad por parte de [Castris](#)

Si necesitas soporte profesional puedes contratar con Castris [soporte profesional](#).

Secuencia de Apple Script para cambiar colores en Numbers

Introducción

Puede que muchos no hallais conocido la época del Apple Script para todo, antes de que Steve Jobs, volviera a su casa y la llevara a donde ha llegado. En aquel entonces el Apple Script era lo mejor para lo makeros frikis. Pro que te permitia cosas impensables en el mundo Windows o en el de linux con las X.

Applescript para funciones de teclado

Para ir más rápido uso un herramienta imprescindible como makero friki: [BetterTouchTool](#) pero puedes apañartelas a pelo. Eso ya es otro cantar que no voya explicar aqui.

Secuencia

En este caso uso 3 secuencias para marcar filas o celdas en numbers, cuando ando con la contabilidad. Numbers es maravilloso, pero algunas cosas no son precisamente usables como esta y prefiero tener un secuencia de teclado para ir mas rápido.

Podeis modificar los colores pero recodsar usar el formato de color de mac

BetterTouchTool

Color verde

```
tell application "Numbers"
  tell front document
    tell active sheet
      try
        set active_table to (first table whose class of selection range is range)
      on error
        display alert "No selection" buttons {"OK"} default button 1
        return
      end try
    tell active_table
```

```
        set background color of selection range to {0, 65535, 0}
    end tell
end tell
end tell
end tell
end tell
```

Color amarillo

```
tell application "Numbers"
    tell front document
        tell active sheet
            try
                set active_table to (first table whose class of selection range is range)
            on error
                display alert "No selection" buttons {"OK"} default button 1
                return
            end try
            tell active_table
                set background color of selection range to {65535, 65535, 0}
            end tell
        end tell
    end tell
end tell
end tell
```

Color rojo

```
tell application "Numbers"
    tell front document
        tell active sheet
            try
                set active_table to (first table whose class of selection range is range)
            on error
                display alert "No selection" buttons {"OK"} default button 1
                return
            end try
            tell active_table
                set background color of selection range to {65535, 17733, 14906}
            end tell
        end tell
    end tell
end tell
```

end tell

Aviso

Esta documentación y su contenido, no implica que funcione en tu caso o determinados casos. También implica que tienes conocimientos sobre lo que trata, y que en cualquier caso tienes copias de seguridad. El contenido el contenido se entrega, tal y como está, sin que ello implique ningún obligación ni responsabilidad por parte de [Castris](#)

Si necesitas soporte profesional puedes contratar con Castris [soporte profesional](#).

Cómo agregar tu firma de correo electrónico HTML en la aplicación Mac Mail.app

Introducción

Años llevamos en Mac Os con esta cuestión que no es entendible en un entorno como el de Apple. Pero allá vamos.

Crear nuestra firma en formato HTML.

Limpia, y a ser posible pasada por algún servicio de validación de HTML para correo electrónico, para evitar incompatibilidades entre programas y sistemas, que son muchas.

La guardaremos en un lugar de nuestro sistema, y la mantendremos abierta para copiar y pegar.

Crear el archivo de firma de correo electrónico en Mail.app

1. Abre Mail.app.
2. Haz clic en "Firma" y luego en el menú desplegable selecciona "Editar firma".
3. Selecciona la bandeja de entrada para la que deseas crear una firma.
4. Presiona el botón "+".
5. Nombra la firma.
6. Cierra la ventana emergente.
7. Cierra completamente Mac Mail.

Paso 2. Localizar el archivo de firma de correo electrónico HTML

1. Abre una ventana del Finder.
2. Selecciona tu carpeta "Macintosh HD".
3. Ve a "Usuarios".
4. Selecciona tu nombre de usuario (si no puedes ver las carpetas ocultas, presiona cmd+shift+. para mostrarlas. Repite esto para ocultarlas de nuevo si lo deseas).
5. Abre "Biblioteca" (una carpeta oculta).
6. Luego abre "Mail".

7. Encuentra y abre "V7" (o la carpeta "V" que sea más reciente).
8. A continuación, abre "MailData".
9. Ahora abre "Signatures".
10. Dentro de esta carpeta, estarán todas tus carpetas de firmas con nombres de archivo aleatorios.
11. Localiza la más reciente (este es el archivo que creaste en el paso 1).

Paso 3. Agregar el código HTML al archivo de Mac Mail

1. Abre el archivo de firma con un editor de texto (preferimos Sublime Text).
2. Deja las primeras 5 líneas de código (la última línea de código comenzará con Mime-Version - deja esta línea completa tal como está).
3. Elimina todas las líneas DEBAJO de esa línea.
4. Copia y pega tu código HTML (desde tu nuevo archivo de firma HTML).
5. Pégallo 2 líneas debajo de la línea Mime-Version.
6. Guarda tu código HTML y cierra el archivo.
7. Vuelve a la ventana del Finder, selecciona el archivo de firma que acabas de editar y presiona Cmd+I (Obtener información).
8. En "General", marca la casilla "bloqueado".
9. Cierra esta ventana y verifica que puedes ver un candado en tu archivo (si alguna vez necesitas editar tu archivo de firma, debes revertir este paso para desbloquear temporalmente el archivo).

Paso 4. Ver tu firma en acción

1. Bueno, ¡si llegaste hasta aquí, felicidades!
2. Abre Mac Mail.
3. Crea un nuevo correo electrónico.
4. Si no puedes ver la firma, asegúrate de que la bandeja de entrada que envía es la que seleccionaste en el paso 2.
5. ¿Aún no la ves? Haz clic en la firma y selecciónala manualmente desde la lista desplegable.
6. ¡Disfruta!

Espero que esto te sea de ayuda. Si necesitas más asistencia, no dudes en preguntar.

Force Sleep en macOS — LaunchAgent con pmset sleepnow

Dejar en suspensión un Mac de no es tan fácil como antes sobre todo si es un equipo estático con discos externos. Discos de backup externos, el micro externo, los altavoces,...

Y él no dejar que descanse tu Mac, le envejece prematuramente. Es como las personas necesitan dormir descansar.

“ Solución para forzar el sleep del Mac ignorando assertions de `bluetoothd`, `powerd` y otros procesos que bloquean el sleep normal.

Variables de configuración

Edita solo este bloque antes de ejecutar cualquier comando.

O usa `sed` para reemplazar en bloque (ver sección al final).

```
MAC_USER="abkrim"           # tu usuario macOS
IDLE_MINUTES=30             # minutos de inactividad antes de dormir
CHECK_INTERVAL=300         # cada cuántos segundos comprueba (300 = 5 min)
SCRIPT_PATH="/Users/$MAC_USER/Library/Scripts/force-sleep.sh"
PLIST_PATH="/Users/$MAC_USER/Library/LaunchAgents/com.user.forcesleep.plist"
LABEL="com.user.forcesleep"
```

El problema

`pmset sleep 30` respeta las **sleep assertions**: procesos como `bluetoothd` o `powerd` pueden decirle al sistema "no te duermas" y el Mac ignora el timer configurado.

```
# Diagnóstico: ver qué está bloqueando el sleep
pmset -g assertions

# El output típico del problema:
# sleep 0 (sleep prevented by bluetoothd, powerd, cloudd...)
```

`pmset sleepnow` **ignora** esas assertions y duerme el Mac sin negociación.

Solución: script + LaunchAgent

1 — Crear el script

```
gral

cat > "$SCRIPT_PATH" << 'EOF'
#!/bin/bash
# Rutas absolutas – los LaunchAgents arrancan sin $PATH normal
IOREG=/usr/sbin/ioreg
PMSET=/usr/bin/pmset
AWK=/usr/bin/awk

IDLE=$(($IOREG -c IOHIDSystem | $AWK '/HIDIdleTime/ {print int($NF/1000000000); exit}')
```

```
# Salir si IDLE está vacío (evita errores de comparación)
[[ -z "$IDLE" ]] && exit 0

if [ "$IDLE" -ge IDLE_SECONDS ]; then
    $PMSET sleepnow
fi
EOF

# Sustituir IDLE_SECONDS con el valor real (30 min = 1800 s)
IDLE_SECONDS=$(( IDLE_MINUTES * 60 ))
sed -i '' "s/IDLE_SECONDS/$IDLE_SECONDS/" "$SCRIPT_PATH"

chmod +x "$SCRIPT_PATH"
```

¿Por qué rutas absolutas?

Los LaunchAgents arrancan con un `$PATH` mínimo (`/usr/bin:/bin`). Sin ruta absoluta, `ioreg` y `pmset` dan **Error 127** (command not found).

2 — Verificar que los binarios existen

```
which ioreg pmset awk
# Esperado:
# /usr/sbin/ioreg
# /usr/bin/pmset
# /usr/bin/awk
```

3 — Crear el plist

“ ⚠ El nombre del archivo **debe** terminar en `.plist`. Sin la extensión o con sufijos extra (ej. `.plist-e`) macOS rechaza el agente con Error 127.

```
cat > "$PLIST_PATH" << EOF
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
  <key>Label</key>
  <string>$LABEL</string>
  <key>ProgramArguments</key>
  <array>
    <string>/bin/bash</string>
    <string>$SCRIPT_PATH</string>
  </array>
  <key>StartInterval</key>
  <integer>$CHECK_INTERVAL</integer>
  <key>RunAtLoad</key>
  <false/>
</dict>
</plist>
```

4 — Cargar el agente

```
launchctl load "$PLIST_PATH"

# Verificar: la segunda columna debe ser 0 (sin error)
launchctl list | grep forcesleep
```

Gestión del agente

```
# Desactivar (sin eliminar)
launchctl unload "$PLIST_PATH"

# Reactivar
launchctl load "$PLIST_PATH"

# Eliminar completamente
launchctl unload "$PLIST_PATH"
rm "$SCRIPT_PATH"
rm "$PLIST_PATH"

# Probar el script manualmente
bash "$SCRIPT_PATH"

# Ver idle time actual (en segundos)
/usr/sbin/ioreg -c IOHIDSystem | /usr/bin/awk '/HIDIdleTime/ {print int($NF/1000000000);
exit}'
```

Diagnóstico de errores comunes

Error	Causa	Solución
Error 127	Binario no encontrado por <code>\$PATH</code> vacío	Usar rutas absolutas en el script

Error	Causa	Solución
Error 127 en LaunchControl	Nombre del plist sin extensión o con sufijo extra	Recrear con nombre exacto *.plist
IDLE vacío	ioreg no devuelve HIDIdleTime	Comprobar which ioreg, verificar ruta
Sleep no ocurre	pmset sleep respeta assertions	Usar pmset sleepnow en su lugar

Adaptar a otro usuario con sed

Para usar esta guía en otra máquina, reemplaza el usuario en bloque:

```
# Reemplazar "abkrim" por el nuevo usuario en todos los comandos
sed 's/abkrim/NUEVO_USUARIO/g' force-sleep-mac.md > force-sleep-mac-custom.md
```

O directamente al ejecutar, exporta las variables al inicio de tu sesión de terminal y copia/pega los bloques de código — las variables \$MAC_USER, \$SCRIPT_PATH, etc. se resolverán solas.

Cómo funciona (resumen)

```
LaunchAgent (cada 5 min)
  |
  ▼
force-sleep.sh
  |
  └─ Lee idle time desde IOHIDSystem (ioreg)
  |
  └─ ¿IDLE >= 1800s? —No→ exit 0
  |
  └─ Sí→ pmset sleepnow ← ignora bluetoothd, powerd, etc.
```

Aviso

Esta documentación y su contenido, no implica que funcione en tu caso o determinados casos. También implica que tienes conocimientos sobre lo que trata, y que en cualquier caso tienes copias de seguridad. El contenido el contenido se entrega, tal y como está, sin que ello implique ningún obligación ni responsabilidad por parte de [Castris](#)

Si necesitas soporte profesional puedes contratar con Castris [soporte profesional](#).

Tips rapidos.

Instalar programas/script python con pip3 en entorno Mac con Homebrew

Muchas veces en mac quieres instalar algo con `pip` pero tu instalación de `pip3` ewsta mantenida por [Homebrew](#) y obtines el error de abajo

```
pip3 install "headroom-ai[all]"
error: externally-managed-environment
```

× This environment is externally managed

↳ To install Python packages system-wide, try `brew install xyz`, where `xyz` is the package you are trying to install.

If you wish to install a Python library that isn't in Homebrew, use a virtual environment:

```
python3 -m venv path/to/venv
source path/to/venv/bin/activate
python3 -m pip install xyz
```

If you wish to install a Python application that isn't in Homebrew, it may be easiest to use `'pipx install xyz'`, which will manage a virtual environment for you. You can install `pipx` with

```
brew install pipx
```

You may restore the old behavior of `pip` by passing the `'--break-system-packages'` flag to `pip`, or by adding `'break-system-packages = true'` to your `pip.conf` file. The latter will permanently disable this error.

If you disable this error, we **STRONGLY** recommend that you additionally

pass the '--user' flag to pip, or set 'user = true' in your pip.conf file. Failure to do this can result in a broken Homebrew installation.

Read more about this behavior here: <<https://peps.python.org/pep-0668/>>

note: If you believe this is a mistake, please contact your Python installation or OS distribution provider. You can override this, at the risk of breaking your Python installation or OS, by passing --break-system-packages.

hint: See PEP 668 for the detailed specification.

Resumen

This environment is externally managed

↳ To install Python packages system-wide, try brew install xyz, where xyz is the package you are trying to install.

Solución

Usar `pipx`

```
brew install pipx
pipx install "headroom-ai[all]"
  installed package headroom-ai 0.26.0, installed using Python 3.14.6
  These apps are now available
    - headroom
done! □ □□□
```

Aviso

Esta documentación y su contenido, no implica que funcione en tu caso o determinados casos. También implica que tienes conocimientos sobre lo que trata, y que en cualquier caso tienes copias de seguridad. El contenido el contenido se entrega, tal y como está, sin que ello implique ningún obligación ni responsabilidad por parte de [Castris](#)

Si necesitas soporte profesional puedes contratar con Castris [soporte profesional](#).