

Lo primero es saber si el procesador del host soporta SSE4.2. Para ello ejecuta en la terminal del ProxmoxVE, lo siguiente:

```
cat /proc/cpuinfo | grep --color=always sse4_2
```

Si aparece sse4_2 en rojo, tienes soporte SSE4.2. Si no, pues no, y aquí acaba este minitutorial para ti.

Una vez comprobado el soporte de SSE4.2 lo siguiente es conseguir el iso de instalación de macOS Catalina. Para ello ejecuta lo siguiente en una terminal de un Mac:

```
curl http://raw.githubusercontent.com/thenickdude/OSX-KVM/master/fetch-macOS.py -o fetch-macOS.py
chmod +x fetch-macOS.py
./fetch-macOS.py
```

Teclea el número correspondiente a la última versión de Catalina y presiona Enter. Espera que se descargue el archivo y después ejecuta:

```
hdiutil convert BaseSystem.dmg -format RdWr -o Catalina-installer.iso
mv Catalina-installer.iso.img CatalinaKVM.iso
```

Una vez que ya dispongas del archivo CatalinaKVM.iso, bájate la última versión de Clover para Catalina desde [aquí](#) y descomprímelo.

Pasa los dos ISOs (tanto el oficial de Catalina como el de Clover) a la carpeta de ISOs de ProxmoxVE (Normalmente **/var/lib/vz/template/iso**)

Ahora sigue las instrucciones de [este hack](#) para obtener la clave OSK de un mac original. Una vez que la obtengas, guárdala.

A continuación crea una máquina virtual de ProxmoxVE con las opciones que veas en la galería de abajo:

Create: Virtual Machine

General OS System Hard Disk CPU Memory Network Confirm

Node: proxmox Resource Pool:

VM ID: 153

Name: catalina

Start at boot:

Start/Shutdown order: any

Startup delay: default

Shutdown timeout: default

Help Advanced Back Next

isos:iso/Catalina-installer-10.15.4-recovery.iso.media=cdrom

Create: Virtual Machine

General **OS** System Hard Disk CPU Memory Network Confirm

Use CD/DVD disc image file (iso)

Storage:

ISO image:

Guest OS:

Type:

Version:

Use physical CD/DVD Drive

Do not use any media

Advanced

Create: Virtual Machine ⓧ

General OS **System** Hard Disk CPU Memory Network Confirm

Graphic card: VMware compatible ▾ SCSI Controller: VirtIO SCSI ▾

Qemu Agent:

BIOS: OVMF (UEFI) ▾ Machine: q35 ▾

Add EFI Disk:

Storage: vms ▾

Format: Raw disk image (raw) ▾

ⓧ Help Advanced Back Next

Create: Virtual Machine

General OS System **Hard Disk** CPU Memory Network Confirm

Bus/Device: SATA 0 Cache: Write back (unsafe)

Storage: vms Discard:

Disk size (GiB): 64

Format: Raw disk image (raw)

SSD emulation: Backup:

IO thread: Skip replication:

Read limit (MB/s): unlimited Read max burst (MB): default

Write limit (MB/s): unlimited Write max burst (MB): default

Read limit (ops/s): unlimited Read max burst (ops): default

Write limit (ops/s): unlimited Write max burst (ops): default

Advanced

Create: Virtual Machine ✕

General
OS
System
Hard Disk
CPU
Memory
Network
Confirm

Sockets:	<input type="text" value="1"/>	Type:	<input type="text" value="Penryn"/>
Cores:	<input type="text" value="4"/>	Total cores:	4
VCPUs:	<input type="text" value="4"/>	CPU units:	<input type="text" value="1024"/>
CPU limit:	<input type="text" value="unlimited"/>	Enable NUMA:	<input type="checkbox"/>

Extra CPU Flags:

Default	- <input type="radio"/> <input checked="" type="radio"/> <input type="radio"/> +	md-clear	Required to let the guest OS know if MDS is mitigated correctly
Default	- <input type="radio"/> <input checked="" type="radio"/> <input type="radio"/> +	pcid	Meltdown fix cost reduction on Westmere, Sandy-, and IvyBridge Intel CPUs
Default	- <input type="radio"/> <input checked="" type="radio"/> <input type="radio"/> +	spec-ctrl	Allows improved Spectre mitigation with Intel CPUs
Default	- <input type="radio"/> <input checked="" type="radio"/> <input type="radio"/> +	ssbd	Protection for "Speculative Store Bypass" for Intel models
Default	- <input type="radio"/> <input checked="" type="radio"/> <input type="radio"/> +	ibpb	Allows improved Spectre mitigation with AMD CPUs
Default	- <input type="radio"/> <input checked="" type="radio"/> <input type="radio"/> +	virt-ssbd	Basis for "Speculative Store Bypass" protection for AMD models

Advanced

Create: Virtual Machine

General OS System Hard Disk CPU **Memory** Network Confirm

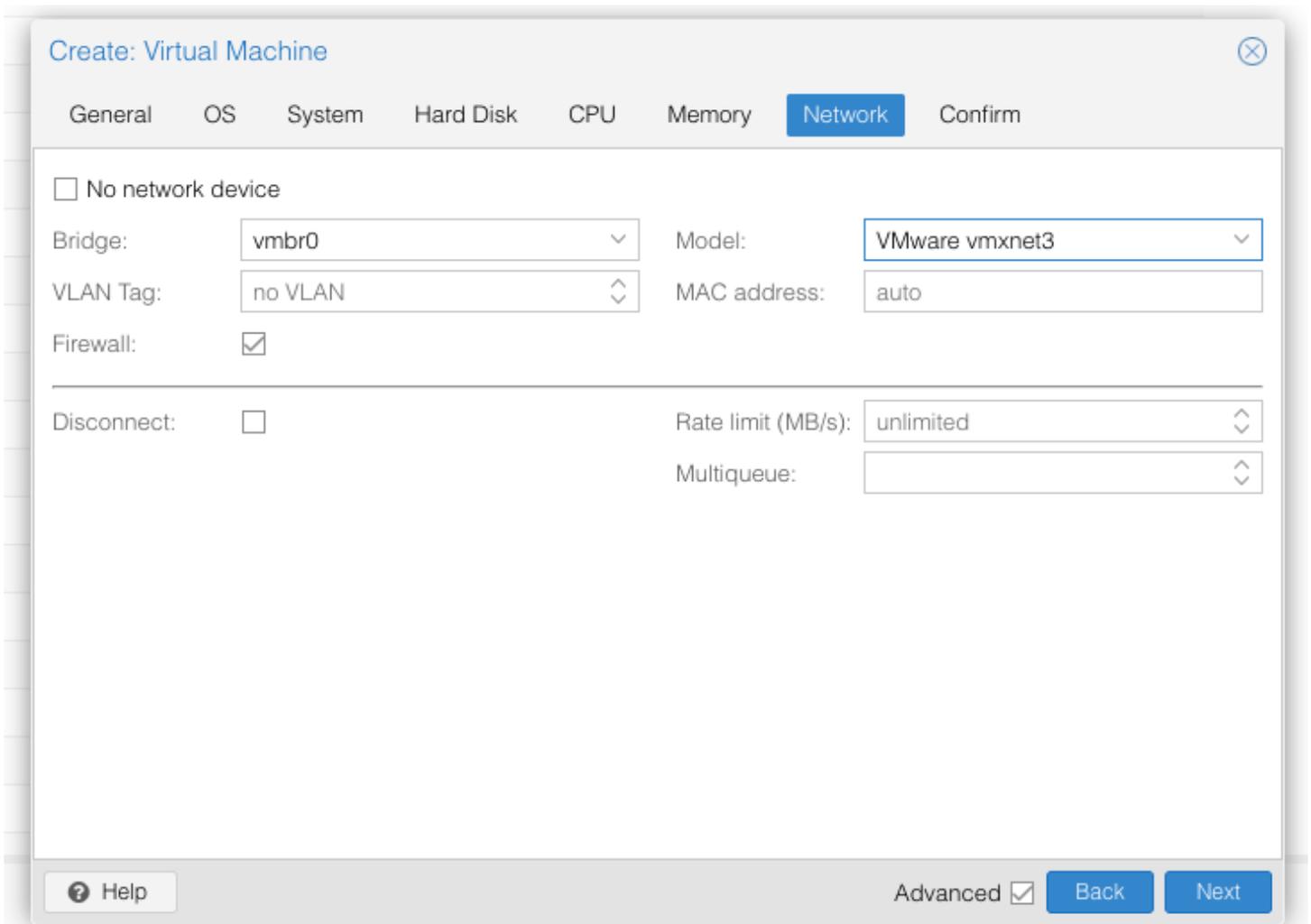
Memory (MiB):

Minimum memory (MiB):

Shares:

Ballooning Device:

[Help](#) Advanced [Back](#) [Next](#)



Al terminar de crear la MV, vete a su pestaña de opciones y asegúrate de que «Usar tableta para el puntero» esté como «Si».

En la pestaña de «Hardware» de la máquina virtual elimina el lector de DVD IDE y agrega uno SATA 0 con el ISO de Clover de antes.

Agrega una segunda unidad de DVD en el SATA 2 e insértale el ISO de Catalina.

Aún sin haber arrancado la máquina virtual de macOS, loguéate por SSH en el host de ProxmoxVE y edita el archivo de configuración de la máquina virtual usando nano. Por ejemplo:

```
nano /etc/pve/qemu-server/103.conf
```

... reemplazando 103 por el ID que le diste a la máquina virtual al crearla.

Una vez que se te abra la edición del archivo, agrega al principio del archivo, la siguiente línea:

```
args: -device isa-applesmc,osk="CLAVEOSK" -smbios type=2 -cpu Penryn,kvm=on,vendor=GenuineIntel,+kvm_pv_unhalt,+kvm_pv_eoi,+invts,vmware-cpuid-freq=on,+pcid,+ssse3,+sse4.2,+popcnt,+avx,+aes,+xsave,+xsaveopt,check -device usb-kbd,bus=ehci.0,port=2
```

... donde CLAVEOSK es la clave larga que obtuviste más arriba.

Ahora, todavía editando el archivo de configuración, busca las líneas que definen los dos ISOs (en este caso sata0 y sata2) y reemplaza «media=cdrom» por «cache-unsafe» en ambas líneas. De esta forma hacemos que ambos ISOs sean tratados como discos duros en vez de como DVDs.

Al terminar el cambio de arriba sal de la edición con «CTRL + x» y acepta guardar los cambios.

Vete otra vez a la pestaña de opciones de la máquina virtual, y en el «Orden de arranque» pon como primero el sata0 que es la unidad donde está insertado el ISO de Clover.

Ahora, para evitar el bucle constante de arranque de macOS en la máquina virtual recién creada, loguéate nuevamente por SSH en el host de ProxmoxVE y ejecuta:

```
echo 1 > /sys/module/kvm/parameters/ignore_msrs
echo "options kvm ignore_msrs=Y" >> /etc/modprobe.d/kvm.conf && update-initramfs -k all -u
```

Ahora, todavía desde el SSH del host de Proxmox, vamos a instalar una versión parcheada de OVMF compatible con macOS (pero que también funciona perfectamente con Windows 10, por si tienes alguna duda). Para ello ejecuta en el host:

```
cd /tmp
wget
http://github.com/thenickdude/pve-edk2-firmware/releases/download/2.20190614-1-macos/pve-edk2-firmware_2.20190614-1_all.deb
dpkg -i /tmp/pve-edk2-firmware_2.20190614-1_all.deb
apt-mark hold pve-edk2-firmware
```

INSTALACIÓN DE CATALINA EN LA MÁQUINA VIRTUAL

- Arranca la máquina virtual e inmediatamente haz doble-click en ella para ver su pantalla. Inmediatamente la veas presiona F2 para acceder a la configuración de OVMF.
- Selecciona «Device manager» y luego «OVMF platform configuration».
- Vete a «Change preferred» y modificala a «1920×1080». Presiona Enter para aceptar el cambio y después F10 para salvarlo. Luego presiona la tecla Escape para volver a la pantalla anterior, otra vez Escape para ir a una pantalla más atrás y luego dale Enter en «Reset» (no en «Continue»). *Todo esto se hace para que la configuración de resolución de Clover coincida con la de OVMF.*
- Una vez hecho lo anterior y habiéndole dado a «Reset», al reiniciar, deberías estar viendo la pantalla de inicio de Clover.
- Dale a «Boot macOS Instal from macOS Base System» y la instalación debería empezar, aunque seguramente tarde un rato en llegar a la pantalla de bienvenida.