

Installing macOS 12 “Monterey” on Proxmox 7

This tutorial for installing macOS 12 Monterey has been adapted for Proxmox from [Kholia's OSX-KVM project](#) and [Leoyzen's OpenCore configuration for KVM](#). You can get the full sourcecode of my OpenCore release [on my GitHub here](#).

Requirements

I'll assume you already have [Proxmox 7](#) installed. You also need a real Mac available in order to fetch the OSK key.

Your Proxmox host computer's CPU must support SSE 4.2, so for Intel your CPU must be at least as new as Nehalem, which was the first CPU generation to bear the “Core” i5/i7 branding. Older CPUs will cause Illegal Instruction crashes when apps/extensions attempt to use these missing instructions.

Modern AMD CPUs also support SSE 4.2 and will work with this guide.

First step: Create an installation ISO

Download my copy of the OSX-KVM repository using the download button, and unzip it:

<https://github.com/thenickdude/OSX-KVM>

First we need to install some build requirements. If you will be building the installer ISO on macOS, open up the Terminal and run this command to install the commandline tools:

```
xcode-select --install
```

If you're building the ISO on Linux, you instead need to run this command (these are the package names for Ubuntu or similar distributions, they may need adjustment on other distributions):

```
sudo apt install qemu-utils make
```

Now in the Terminal, from the root of OSX-KVM, run:

```
cd scripts/monterey  
make Monterey-recovery.img
```

This will download the Monterey installer from Apple's software distribution servers and build a `Monterey-recovery.img` file for you. Upload this file to your Proxmox's ISO store directory (typically

/var/lib/vz/template/iso). Although we're putting it in the ISO directory so that we can use it with Proxmox's ISO picker later, this is a raw disk image rather than a true ISO.

If you're building the installer on macOS, you can build a full installer instead of just a recovery, which will mean that macOS won't have to download Monterey files during installation, and so won't require an Internet connection. Simply ask it to build Monterey-full.img instead:

```
cd scripts/monterey
make Monterey-full.img
```

This option is not available when building the installer on Linux.

Prepare an OpenCore image

Download the [OpenCore.iso.gz file](#) from the newest release in my repository (you want v15 or newer), double click it to unpack it, and upload it to Proxmox's ISO store at /var/lib/vz/template/iso. Although it has a .iso file extension, this is actually a hard disk image.

Fetch the OSK authentication key

macOS checks that it is running on real Mac hardware, and refuses to boot on third-party hardware. You can get around this by reading an authentication key out of your real Mac hardware (the OSK key). Save the first block of C code from [this page](#) as smc_read.c. In a command prompt, change into the same directory as that file and run:

```
xcode-select --install # If you don't already have gcc
gcc -o smc_read smc_read.c -framework IOKit
./smc_read
```

It'll print out the 64 character OSK for you. Make a note of it.

Every Mac uses the same OSK, so don't be surprised that it doesn't look like a random string!

Create the VM

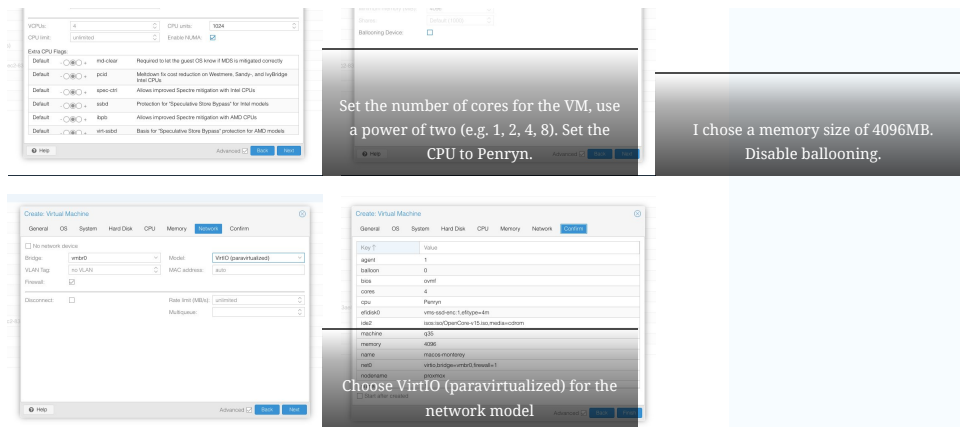
From the Proxmox web UI, create a new virtual machine as shown below.

Keep a note of your VM's ID

Select the OpenCore ISO you uploaded and set OS type to "Other"

Set graphics to "VMWare Compatible", set BIOS to "VMware (UEFI)", set Machine to "Q35", tick QEMU Agent, tick Add EFI Disk and pick storage for it. You must untick "pre-enroll keys"

Set the size of the hard disk (64GB or greater, 32GB is too small). Attach it to virtio0. Enable discard to support TRIM.



In the Options page for the VM, ensure that “use tablet for pointer” is set to “Yes” (this is the default).

In the Hardware page for the VM, add a second DVD drive at IDE0, set it to use your Monterey-full.img or Monterey-recovery.img.

Don’t try to start the VM just yet. First, SSH into your Proxmox server so we can make some edits to the configuration files.

Edit /etc/pve/qemu-server/YOUR-VM-ID-HERE.conf (e.g. with nano or vim). Add this line, being sure to substitute the OSK you extracted earlier into the right place:

```
args: -device isa-applesmc,osk="THE-OSK-YOU-EXTRACTED-GOES-HERE"
      -smbios type=2 -device usb-kbd,bus=ehci.0,port=2 -global nec-usb-xhci.msi=off -global ICH9-LPC.acpi-pci-hotplug-with-bridge-support=off
```

A USB keyboard is added here because macOS doesn’t support QEMU’s default PS/2 keyboard.

MSI has been disabled for the USB controller to fix QEMU panics when USB 3 devices are passed through.

ACPI PCIe hotplug support needs to be disabled for VMs with passthrough devices since QEMU Q35 machine model 6.1, which is what that final argument does. (You could instead choose Q35 6.0 for your Machine type on the Hardware tab). If you haven’t updated your QEMU to version 6.1 yet then remove this argument from the args. (Proxmox 7 ships with 6.0 and updated to 6.1 later on)

Ensure the args are all on a single line!

We also need to add a -cpu argument. **If your host CPU is Intel**, add this to the end of the “args” line:

```
-cpu
host,kvm=on,vendor=GenuineIntel,+kvm_pv_unhalt,+kvm_pv_eoi,+hypervisor,+invtsc
```

This will pass through all of the features that your CPU supports. OpenCore’s config will pretend to macOS that the CPU’s model name is Penryn for compatibility.

If your host CPU is AMD, or the above argument doesn’t work for you, use this more-compatible alternative:

```
-cpu
Penryn,kvm=on,vendor=GenuineIntel,+kvm_pv_unhalt,+kvm_pv_eoi,+hypervisor,+invtsc,+pcid,+ssse3,+sse4.2,+popcnt,+avx,+avx2,+aes,+fma4,+fma3,+bmi1,+bmi2,+xsave,+xsaveopt,+rdrand,check
```

This pretends that your CPU is Penryn, which will keep macOS happy even if your host CPU is AMD, and adds a bunch of newer required and optional CPU features on top. Features that your host CPU doesn't support will be ignored (a warning will be printed to the console during launch with `qm start 1xx`), but note that macOS won't work without SSE4.2 support.

You can remove the “+invts” feature from the list if your CPU doesn't support it, or if you want to be able to migrate a running VM between Proxmox nodes.

Now find the lines that define the two “ISOs” (ide0 and ide2), and remove the “,media=cdrom” part from them. Add “,cache=unsafe” in its place. This will treat these as hard disks rather than DVD drives.

Save your changes, return to the Options tab, and change the boot order to put IDE2 (the OpenCore image) first. Your final VM configuration file should resemble this:

```
args: -device isa-applesmc,osk=..." -smbios type=2 -device usb-kbd,bus=ehci.0,port=2 -cpu host,kvm=on,vendor=GenuineIntel,+kvm_pv_unhalt,+kvm_pv_eoi,+hypervisor,+invts -global nec-usb-xhci.msi=off -global ICH9-LPC.acpi-pci-hotplug-with-bridge-support=off
agent: 1
balloon: 0
bios: ovmf
boot: order=ide2
cores: 4
cpu: Penryn
efidisk0: vms-ssd-enc:vm-171-disk-1,efitype=4m,size=1M
ide0: isos:iso/Monterey-full.img,cache=unsafe,size=14G
ide2: isos:iso/OpenCore-v15.img,cache=unsafe,size=150M
machine: q35
memory: 4096
name: macos-monterey
net0: virtio=...,bridge=vbr0,firewall=1
numa: 1
ostype: other
virtio0: vms:vm-171-disk-0,cache=unsafe,discard=on,size=64G
scsihw: virtio-scsi-pci
smbios1: uuid=...
sockets: 1
vga: vmware
```

Configure Proxmox

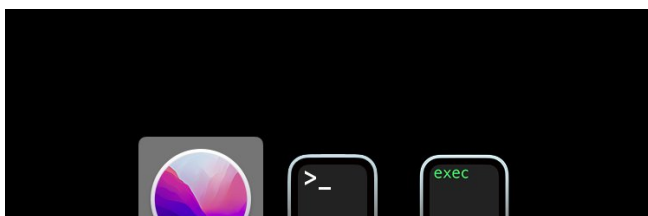
On Proxmox, run “`echo 1 > /sys/module/kvm/parameters/ignore_msrs`” to avoid a bootloop during macOS boot. To make this change persist across Proxmox reboots, run:

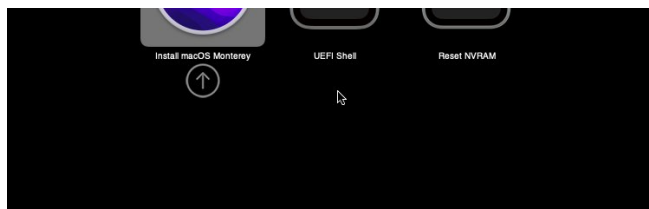
```
echo "options kvm ignore_msrs=Y" >> /etc/modprobe.d/kvm.conf &&
update-initramfs -k all -u
```

Install Monterey

Now start up your VM, it should boot to the OpenCore boot picker:

Press enter to boot the “Install macOS Monterey” entry and the installer should appear. (If your keyboard isn't working, leave the Proxmox Console page and re-enter it)

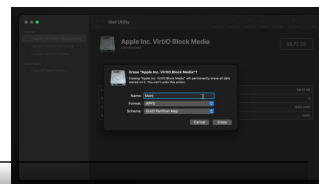
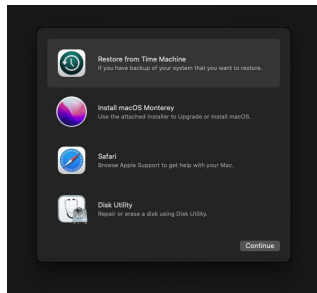




OpenCore's "OpenCanopy" boot picker

If you built a recovery installer, the icon will instead be an image of a hard disk and be labelled "MacOS Base System".

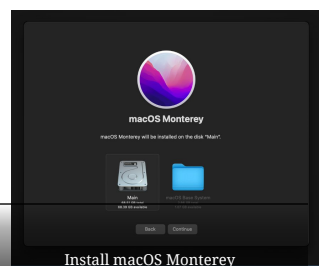
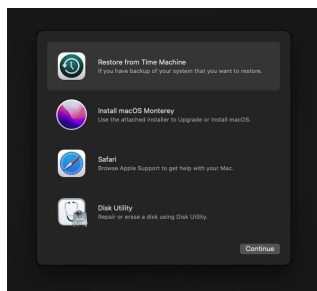
Our virtual hard drive needs to be erased/formatted before we can install to it, so select the Disk Utility option. Follow the steps below to format the disk:



Select Disk Utility

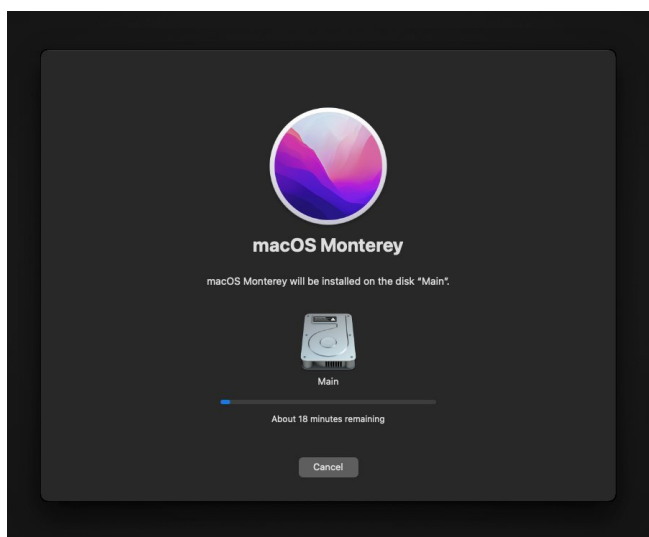
Erase the main disk (it'll appear as VirtIO Block Media). Choose APFS. I called mine "Main"

Now you can quit Disk Utility from the top menu (Disk Utility > Quit Disk Utility), and we're ready to begin installation!



Install macOS Monterey

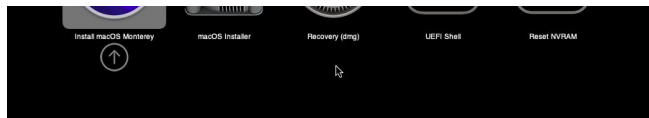
Choose the disk you erased



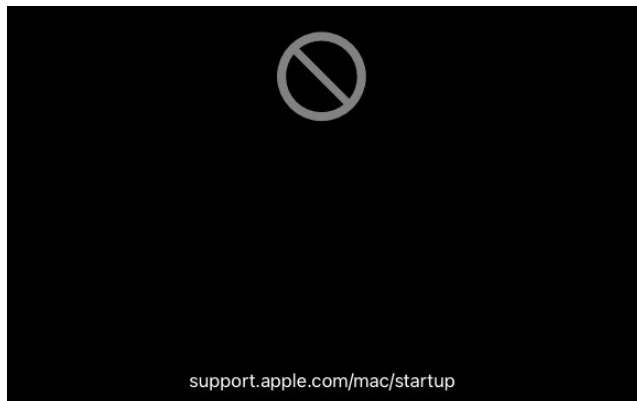
It's installing!

After the first stage of installation, the VM will reboot 2 or 3 times in quick succession, and each time you must manually pick the "macOS Installer" entry (the second one here, with the hard disk icon) to continue installation. It will not be selected for you automatically:





If your keyboard isn't responding on this screen, exit the Console tab in Proxmox and re-enter it. If you get a "prohibited" sign like this appearing, hit the Reset button on the VM to try again:



Now the installation is nearly complete and the macOS Installer entry disappears, so pick the name of your main disk to boot (mine's called Main).



There's just one more reboot to come. Afterwards pick the "Main" entry again and this time you'll finally boot into Monterey!

Answer the initial install questions, and you'll be logged on! Note that you will want to hold off on logging into your Apple ID until you've configured your Mac's serial number in OpenCore (because otherwise a Mac with the default shared serial number in my OpenCore image will be added to your Apple ID).



It works!

Note that it will be really sluggish for a few minutes after the first boot while the system performs housekeeping tasks.

Make the OpenCore install permanent

We're currently booting using OpenCore from the attached OpenCore

ISO. Let's install that to the hard drive instead. Pop open Terminal and run "diskutil list" to see what drives we have available.

```
nick -- zsh -- 83x35
Last login: Tue Oct 26 13:18:14 on console
nick@iMac-Pro ~ % diskutil list
/dev/disk0 (internal):
#    TYPE NAME             SIZE IDENTIFIER
0:    GUID_partition_scheme      68.7 GB disk0
1:      EFI EFI              209.7 MB disk0s1
2:      Apple_APFS Container disk3  68.5 GB disk0s2

/dev/disk1 (internal, physical):
#    TYPE NAME             SIZE IDENTIFIER
0:    GUID_partition_scheme     *157.3 MB disk1
1:      EFI EFI              157.2 MB disk1s1

/dev/disk2 (internal, physical):
#    TYPE NAME             SIZE IDENTIFIER
0:    GUID_partition_scheme     *15.0 GB disk2
1:      EFI EFI              209.7 MB disk2s1
2:      Apple_HFS Install macOS Monterey 14.7 GB disk2s2

/dev/disk3 (synthesized):
#    TYPE NAME             SIZE IDENTIFIER
0:    APFS Container Scheme -   +68.5 GB disk3
   Physical Store disk0s2
1:      APFS Volume Main - Data    1.8 GB disk3s1
2:      APFS Volume Preboot       269.7 MB disk3s2
3:      APFS Volume Recovery       1.1 GB disk3s3
4:      APFS Volume VM            1.1 MB disk3s4
5:      APFS Volume Main          15.7 GB disk3s5
6:      APFS Snapshot com.apple.os.update-... 15.7 GB disk3s5s1

nick@iMac-Pro ~ %
```

Use "sudo dd if=<source> of=<dest>" to copy the "EFI" partition from the OpenCore CD and overwrite the EFI partition on the hard disk. The OpenCore CD is the small disk (~150MB) that only has an EFI partition on it, and the main hard disk is the one with the large (>30GB) Apple_APFS "Container" partition on it.

In my case these EFI partitions ended up being called disk1s1 and disk0s1 respectively, so I ran "sudo dd if=/dev/disk1s1 of=/dev/disk0s1" (note that if you get these names wrong, you will overwrite the wrong disk and you'll have to start the installation over again!).

Now shut down the VM, and remove both the OpenCore and the Monterey installer drives from the Hardware tab. On the Options tab, edit the boot order to place your virtio0 disk as the first disk. Boot up. If everything went well, you should see the OpenCore boot menu, and you can select your "Main" disk to boot Monterey:



Sleep management

I found that I was unable to wake Monterey from sleep using my mouse or keyboard. If you encounter the same problem, you can either disable system sleep in Monterey's Energy Saver settings to avoid the issue, or you can manually wake the VM up from sleep from Proxmox by running:

```
qm monitor YOUR-VM-ID-HERE
system_wakeup
quit
```

Editing your OpenCore/EFI settings

The Configuration.pdf that explains the OpenCore config.plist file can be found along [with the OpenCore release on my GitHub](#).

To mount your EFI partition in macOS so you can edit your config.plist, first check the “identifier” of your EFI partition in the terminal:

```
~$ diskutil list
/dev/disk0 (external, physical):
#:

|            | TYPE                       | NAME | SIZE      |         |
|------------|----------------------------|------|-----------|---------|
| IDENTIFIER |                            |      |           |         |
| 0:         | GUID_partition_scheme      |      | *512.1 GB | disk0   |
| 1:         | EFI                        | EFI  | 209.7 MB  | disk0s1 |
| 2:         | Apple_APFS Container disk1 |      | 511.9 GB  | disk0s2 |


```

Then you can mount it like so:

```
sudo mkdir /Volumes/EFI
sudo mount -t msdos /dev/disk0s1 /Volumes/EFI
```

Now you can edit /Volumes/EFI/OC/config.plist with your favourite text editor to make your changes. (TextEdit is not a great choice because it likes to insert curly quotes into the file and otherwise break things, there are some dedicated plist editors available such as XCode).

If you’re unable to boot macOS, you can edit the config.plist using the “UEFI Shell” option in the OpenCore boot menu instead.

Enter “FS0:” and press enter to open up the first available filesystem, then run “edit EFI\OC\config.plist” (if the file isn’t found, try switching to another filesystem like fs1:). When you’re done editing, press control+Q to exit, “Y” to save, then run “exit” to return to the OpenCore menu. You need to reboot for your changes to take effect.

If you prefer, you can edit config.plist from the comfort of your Proxmox host instead. If you’re booting from an attached OpenCore.img file, you can mount that file as a disk on the host. If you’re booting from the VM’s disk instead, it must be in raw format in order to be mounted (e.g. typical LVM or ZFS usage) rather than qcow2.

```
# Mount an OpenCore image:
losetup --partscan /dev/loop0 /var/lib/vz/template/iso/OpenCore-v15.img
# or a VM boot disk:
losetup --partscan /dev/loop0 /dev/zvol/tank/vms/vm-100-disk-1

mount /dev/loop0p1 /mnt
```

Now the contents of that first partition are available in /mnt, so you can edit /mnt/EFI/OC/config.plist in your favourite editor. When you’re done, do this to unmount the disk:

```
umount /mnt
losetup --detach /dev/loop0
```

Automatic boot

In config.plist, you can set Misc/Boot/Timeout to a non-zero value to allow the default boot option be chosen automatically after that delay in seconds. I’ve disabled this by default because it causes the installer ISO to re-enter its main menu instead of continuing the second stage of installation.

You can set the default boot option by pressing control+enter on it.

Verbose boot

To boot macOS in Verbose mode to diagnose boot problems, at the OpenCore boot menu press Cmd+V before pressing enter to boot macOS (you don't need to hold it down).

If there is a kernel panic during boot and it reboots too quickly to be read, edit config.plist to add “debug=0x100” to the kernel arguments.

Changing screen resolution

To change macOS' screen resolution, you need to edit the UEFI/Output/Resolution setting in config.plist, the default is 1920×1080@32.

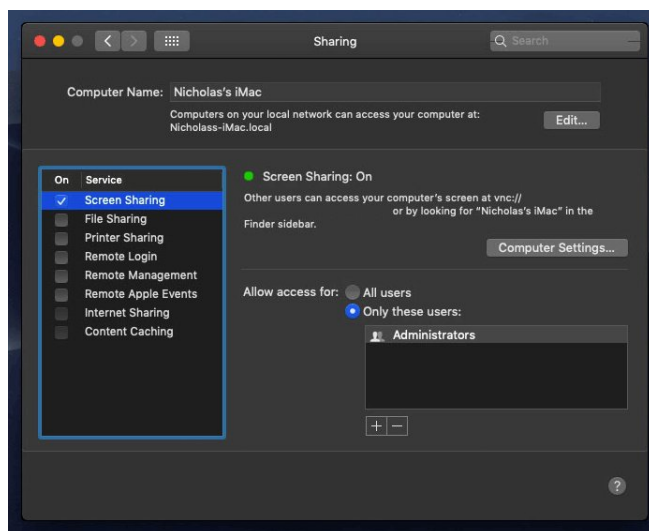
You should be able to change this to any of the modes that the system OVMF menu offers (hit F2 at the start of guest boot and choose “Device Manager/OVMF Platform Configuration” to see which resolutions are available).

Video performance

Because there is no guest video acceleration available for macOS, video performance is poor.

In Google Chrome in the guest you will need to toggle off the setting to “use hardware acceleration when available” to improve issues with elements not being drawn or flickering (especially video). Safari may be a better choice.

macOS's built in “Screen Sharing” feature offers dramatically better framerates and latency than Proxmox's browser-based VNC console, so if you have a real Mac to act as a viewing console, you can enable that in the VM's “Sharing” settings and connect to the VM using the Screen Sharing app from your Mac instead:



Apparently Screen Sharing is also compatible with VNC clients like RealVNC, so you should be able to connect to it from Linux or Windows consoles using RealVNC.

The real magic bullet for video performance is to pass through a compatible video card using PCIe passthrough (though note that macOS now only supports a handful of very old NVidia cards). This offers near-native performance. You can read more about how I'm using PCIe passthrough on my own installation [here](#).

USB passthrough

Since I want to use this as my primary computer, I want to use a USB keyboard and mouse plugged directly into Proxmox, rather than sending my input through the web VNC console.

[Proxmox has good documentation](#) for USB passthrough. Basically, run “qm monitor YOUR-VM-ID-HERE”, then “info usbhost” to get a list of the USB devices connected to Proxmox:

```
qm> info usbhost
Bus 3, Addr 12, Port 6, Speed 480 Mb/s
Class 00: USB device 8564:1000, Mass Storage Device
Bus 3, Addr 11, Port 5.4, Speed 12 Mb/s
Class 00: USB device 04d9:0141, USB Keyboard
Bus 3, Addr 10, Port 5.1.2, Speed 12 Mb/s
Class 00: USB device 046d:c52b, USB Receiver
Bus 3, Addr 9, Port 14.4, Speed 12 Mb/s
Class 00: USB device 046d:c227, G15 GamePanel LCD
Bus 3, Addr 8, Port 14.1, Speed 1.5 Mb/s
Class 00: USB device 046d:c226, G15 Gaming Keyboard
```

In this case I can add my keyboard and mouse to USB passthrough by quitting qm, then running:

```
qm set YOUR-VM-ID-HERE -usb1 host=04d9:0141
qm set YOUR-VM-ID-HERE -usb2 host=046d:c52b
```

This saves the devices to the VM configuration for you. You need to reboot to have the new settings apply.

You can also pass through USB devices by passing through an entire USB controller using Proxmox’s PCIe passthrough feature, which gives much better compatibility.

Odd core counts

If your host CPU is AMD, you might have a core count which is divisible by 3, like 6 cores. macOS doesn’t like it if the core count isn’t a power of 2 (1, 2, 4, 8), but it doesn’t mind odd socket counts.

So in order to pass 6 total cores, configure the VM for 3 sockets and 2 cores per socket.

Fixing “guest boots to UEFI shell”

If your guest ends up booting to the UEFI shell instead of showing the OpenCore boot menu, especially if you’ve just updated OpenCore to a new version, you’ll need to edit the guest’s UEFI boot entries to fix this.

At the very start of guest boot, hit F2 to enter guest UEFI settings.

First we’ll remove the old entries. Choose the Boot Maintenance option, then Boot Options -> Delete Boot Option. Use the spacebar to tick any old Clover or OpenCore entries (avoid ticking the EFI Internal Shell option, you want to keep that!). Select “Commit Changes and Exit”.

Now we’ll add the correct entry for OpenCore back in. Select Add Boot Option. Navigate through the device tree to EFI/OC/OpenCore.efi and select it, name this new option “OpenCore” or similar. Again Commit Changes and Exit.

Go to the Change Boot Order and move OpenCore to the top. Commit

Changes and Exit.

Now back out to the main menu and choose Reset, and you should successfully boot into OpenCore this time.

Fixing iMessage

iCloud and the App Store should already be working for you, but for iMessage support you must follow these steps to mark your network adapter as built-in:

<https://dortania.github.io/OpenCore-Post-Install/universal/iservices.html#fixing-en0>

Disabling SIP (System Integrity Protection)

You can disable SIP by selecting the Recovery option from the OpenCore boot menu, then use the top menu to open the Terminal and run `csrutil disable --no-internal`. Then reboot.

This may be needed to run unsigned kexts or perform other hacks.

Upgrading OpenCore

Sometimes you need to update OpenCore to a new release in order to support a new macOS update. I'll assume you don't have any customisations to config.plist you want to save.

First take a snapshot! It's great to be able to roll back if something goes wrong.

If you're still able to boot macOS, you can update it from within the guest. Follow the instructions in the "Editing your OpenCore/EFI settings" section to mount your EFI partition. Then you can delete the EFI folder in there and replace it with the one from the [OpenCoreEFIFolder.zip file from my OpenCore release](#) (you'll probably need to empty the trash first to make room for the new folder). You're done!

If you aren't able to boot macOS, unpack and upload the [new OpenCore ISO](#) to Proxmox's ISO store instead. Add a new CD drive to the VM that uses that ISO. Then in Proxmox's terminal edit the VM's config (in `/etc/pve/qemu-server`) to replace "media=cdrom" with "cache=unsafe" for the OpenCore drive. Now on the "Options" tab, change the boot order to put the new OpenCore drive first.

Start the VM and boot into macOS using the new OpenCore drive. From within macOS you can now follow the instructions from the "Make the OpenCore install permanent" section to install the new OpenCore image to your main macOS disk, after which the OpenCore drive can be detached from the VM.

Upgrading from macOS Big Sur

First make a backup or snapshot of your system! Being able to roll back when the upgrade goes wrong is a real lifesaver.

You'll need to update OpenCore to my v15 release before the upgrade. You can follow the instructions in the "Editing your OpenCore settings"

section above to mount your EFI disk. Then you can replace the OpenCore files in the mounted “EFI” disk with the ones from the OpenCoreEFIFolder.zip file in my [newest OpenCore release](#).

Reboot to make sure that you can still boot Big Sir.

If you’re using any PCIe passthrough devices (particularly video cards) you’ll want to disable those and set “vga: vmware” instead, so you can install using Proxmox’s web console from a different machine during the upgrade. This avoids installer problems triggered by flaky video card passthrough, especially host lockups caused by the AMD Reset Bug.

Now you can upgrade to Monterey using Software Update or from the App Store like you would on a real Mac.

If the update fails to appear, your SIP mode might be set to “Apple Internal”. You can check this by running “csrutil status” in the Terminal.

To fix this, reboot your VM and at the OpenCore menu select Recovery. Open up the Terminal, and in there run “csrutil enable --no-internal” (“csrutil disable --no-internal” is fine too), then reboot back into macOS and try again.

📅 October 26, 2021 👤 Nicholas Sherlock 📁 macOS / Hackintosh, Proxmox

670 thoughts on “Installing macOS 12 “Monterey” on Proxmox 7”

← OLDER COMMENTS



Krish

June 19, 2022 at 5:10 pm

Proxmox version: 7.2-3

Getting following error while installing mac-os

<https://i.imgur.com/T8a8NsC.png>

error log: “will not load trust cache because required files are missing”

Could you please help me move to this error

REPLY



Incs

June 19, 2022 at 6:59 pm

Hi, thanks for this great description.

I have set up Monterey with GPU Passthrough on Proxmox and it is working fine.

Just one thing is annoying. When I activate Start on boot in Proxmox options it will not work. Screen stays dark and I have to use my Laptop to restart the VM. After Restart Monterey boots and works fine.

Any Idea how to solve this?

REPLY



Nicholas Sherlock

June 19, 2022 at 9:23 pm

That's weird, check dmesg for errors about passthrough.

REPLY



Incs

June 20, 2022 at 9:35 pm

I am not very experienced with Linux.
I made the check with following command and get this result:
dmesg | grep -i "error"
[1.459990] RAS: Correctable Errors collector initialized.
[6.876127] EXT4-fs (dm-1): re-mounted. Opts: errors=remount-ro.
Quota mode: none.
[7.085793] platform regulatory.0: Direct firmware load for
regulatory.db failed with error -2
[7.161414] b43-phy1 ERROR: FOUND UNSUPPORTED PHY (Analog
12, Type 11 (AC), Revision 1)
[7.161496] b43: probe of bcma0:1 failed with error -95
[8.105034] nvidia-gpu 0000:41:00.3: i2c timeout error e0000000
[8.105219] ucsi_ccg: probe of 0-0008 failed with error -110

Is there a better way to check with dmesg?

REPLY



Nicholas Sherlock

June 20, 2022 at 10:48 pm

Mentioned in there is nvidia-gpu, the Nvidia host driver. Ideally
you would be assigning your GPU to vfio-pci on boot before the
Nvidia driver loads, so that it never gets used.

Did you assign your GPU to vfio-pci in your modprobe config? If
not, find a guide for that.

REPLY



Nicholas Sherlock

June 20, 2022 at 3:45 am

There seems to currently be a problem in the data that macOS
downloads during installation if you choose the Recovery method for
installing Monterey. Installation fails with this message:

<https://i.imgur.com/vFLA9DT.png>

Even using working OpenCore and Recovery images from 6 months ago
also replicates the problem, so it has to be an issue with the new data the
Recovery is downloading.

I'm investigating why that's happening

REPLY



Streexz

June 23, 2022 at 2:43 pm

Hi ! Resolved ?

REPLY



Nicholas Sherlock

June 23, 2022 at 9:36 pm

Not yet

REPLY



Streexz

June 24, 2022 at 10:39 am

I find a temporary solution, install Big Sur Bigsur-recovery.img then update it to Monterey

REPLY



Krish

June 21, 2022 at 4:42 pm

Hey, Someone help me to get me out from this error
Every time I stuck at this point

Log: <https://i.imgur.com/A603nEi.png>

Thanks

REPLY



Luis

June 22, 2022 at 8:42 am

I get the error:

The file "bootbase.efi" couldn't be opened because there is no such file.

How to fix that problem?

REPLY



tony

June 25, 2022 at 7:30 pm

I too had the error about bootbase.efi not being found and it stalled out.

I believe this is a problem that is common to trying to upgrade unsupported Macs that have had their original SSDs replaced and the installer is trying to upgrade some firmware component that doesn't pass validation, but I don't know if this 100% the case. In the situation of the real macs, simply having an original SSD detected somehow gets past this situation but not everyone had their original SSD on hand.

Regardless, I managed to bypass this issue by seeing that this problem only started recently, likely started with Monterey 12.4.

I managed to mangle my way to get the full 12.3.1 installer downloaded (Mr. Macintosh or MDS) and pushed into the proxmox ISO directory (16GB). It installed successfully and things are looking OK, and now I'm walking through the next steps.

MacOS software update installed 12.4 successfully for me after that. Looks like it crashed the entire proxmox system during the update

but I had a similar crash installing Windows 11 too, but now after a couple of VM reboots everything looks OK.

Also I have 12.5 beta 3 on deck to see if Apple has addressed this installation issue but now that I went 12.3.1 -> 12.4 successfully I'm going to leave it for now.

REPLY



ben

June 26, 2022 at 1:47 pm

I have the same problem 😞

REPLY



Florian

June 22, 2022 at 12:34 pm

Hi Nick,

When I try to run the installer, I get this image and nothing happens anymore:

<https://share.zkr.io/img/Tw4ik.png>

Did I configure something wrong?

Thanks!

REPLY



Verratti

June 22, 2022 at 2:36 pm

An Error Occurred while preparing the installation. Try running this application again...

REPLY



Krish

June 23, 2022 at 4:30 pm

I got the same error , If you found any solution Please Ping me

REPLY



Patlo

June 27, 2022 at 8:20 pm

Did you find a solution to this error? I'm coming up with the same thing

REPLY



Brad

June 22, 2022 at 5:24 pm

Can you elaborate on passing through usb host controller pci devices. Simply adding them in proxmox doesn't seem to work. I assume there's something more to be done in OpenCore like for the sound chip, but everything I search for just shows how to do single usb devices and only mentions you can do it the other way without giving any instructions.

Also , I'm having an issue that I can get video passthrough to work, but I cannot boot if I remove the VMWARE video option. Anyone have a fix? I am using an nvidia 650ti (Which hasn't had any other problems , supposedly a memory leak has made them unstable for other people) (That's on Big Sur by the way,)

REPLY



Khai

June 23, 2022 at 2:37 am

Gigabyte X570 Aorus , Ryzen 9 3900x ,16GB Ram. I followed everything to the latter. But after running the VM, it just stuck/freeze at Apple Logo, there is no progress bar displayed here. Anyone having same problem? What is the solution(s). Thanks

REPLY



Krish

June 25, 2022 at 3:41 pm

Getting same error i7-12700f RTX3050

REPLY



Krish

July 2, 2022 at 2:13 pm

I Fixed my error to change the CPU cores
My CPU(i7-12700f) have 12 cores but its stuck on Apple logo than I set to 8 cores than its working fine
So try to change the cores number

REPLY



Daniel

August 1, 2022 at 6:57 pm

Same here.

REPLY



Luis

June 23, 2022 at 8:26 am

Hey,

does anyone know, why the graphic is so slow?
Do you really need a strong graphics card or can you use Remote Desktop as with Windows servers?

REPLY



Nicholas Sherlock ▲

June 23, 2022 at 11:13 am

That's because there is no video acceleration available.

You can use macOS's built-in screen sharing feature, it's much faster than the Proxmox Web UI.

But the only way to get proper 2D/3D acceleration is to PCIe

passthrough a compatible GPU

REPLY



Luis

June 23, 2022 at 11:50 am

“You can use macOS’s built-in screen sharing feature, it’s much faster than the Proxmox Web UI.”

But how can I connect to mac like RDP? And where can I activate it?

REPLY



Nicholas Sherlock

June 23, 2022 at 12:09 pm

System Preferences, the Sharing button.

You may be able to connect to it with a generic VNC client.

REPLY



Nicholas Sherlock

June 28, 2022 at 6:54 am

I’ve narrowed down why the Recovery installer is broken right now, it looks like it’s accidentally downloading the new 21F2092 build of macOS during install, which is for M2 Macs.

Seeing if I can fix this now

REPLY



Packersowner

June 28, 2022 at 4:20 pm

Thanks for the guide. I am using a Ryzen 3900X and I have found that it does not like the cpu args: +pcid and +fma4 so I left them out. Also, I can’t seem to get my GPU to passthru without enabling the display=Vmware compatible. Also is the normal warning at startup: no efidisk configured! Using temporary efivars disk.

REPLY



Moses

June 30, 2022 at 3:30 am

Hi Nick thanks for all you do – any ETA on this fix?

“it looks like it’s accidentally downloading the new 21F2092 build of macOS”

REPLY



Nicholas Sherlock

June 30, 2022 at 4:24 am

If you want a quicker fix, you can use my previous tutorial to install a Big Sur VM, then use that VM to build a Full Installer for Monterey. The Full Installer isn’t affected by this problem.

REPLY



Moses

June 30, 2022 at 12:55 pm

Update: like other commentators have mentioned, I was able to get Monterey going by installing Big Sur and upgrading to Monterey. Thanks!

REPLY



Jack.J.X

July 1, 2022 at 6:54 am

For anyone who has the smc_read execution issue which shows error “smc_read.c:24:56: warning: ‘kIOMasterPortDefault’ is deprecated: first deprecated in macOS 12.0 [-Wdeprecated-declarations]”, you have to modify “kIOMasterPortDefault” to “kIOMainPortDefault” according to

<https://github.com/kovidgoyal/kitty/issues/3719>

REPLY



Jack.J.X

July 1, 2022 at 6:58 am

damn seems it does not work anymore, after I execute the smc_read, it shows:

“...” [removed – nick]

Monterey 12.4

REPLY



Nicholas Sherlock

July 1, 2022 at 6:58 am

That's the correct key.

REPLY



Alper

July 5, 2022 at 12:51 pm

Hi, I output lspci on Nick Video Card as follows. Under normal conditions, I get the Gpu transition with the Nvidia GT630 Graphic Card without any problems. However, I could not switch the gpu with the following card.

My Card lspci proxmox output.

05:00.0 VGA compatible controller [0300]: Advanced Micro Devices, Inc. [AMD/ATI] Cape Verde PRO / Venus LE / Tropo PRO-L [Radeon HD 8830M / R7 250 / R7 M465X] [1002:6... (rev 87)]

My Card Real Name ; Dell R7 450

<https://www.techpowerup.com/gpu-specs/dell-r7-450-4-gb.b4966>

I did the procedures according to the steps below, but somehow I could not boot with the proxmox vm mac os new amd video card

<https://dortania.github.io/Getting-Started-With->

[ACPI/Universal/spoof.html](#)

<https://pci-ids.ucw.cz/read/PC/1002/683d>

I enter the following path as the device path.

PciRoot(0x0)/Pci(0x1C,0x0)/Pci(0x0,0x0)

I enter the presentation as the fake id.

683dCape Verde XT [Radeon HD 7770/8760 / R7 250X]
3D680000

I add ssdt-gpu-spoof.aml to the acpi part with opencore. In the device properties, I enter in "PciRoot(0x0)/Pci(0x1C,0x0)/Pci(0x0,0x0) in the device section, but the mac os does not boot. The screen goes black and interestingly, the proxmox server resets completely.

I need your help on this. Where could I be doing wrong? Am I entering the Device Path incorrectly? Or is this related to proxmox? Located on the following port on the Proxmox server

05:00.0 VGA compatible controller [0300]: Advanced Micro Devices, Inc. [AMD/ATI] Cape Verde PRO / Venus LE / Tropo PRO-L [Radeon HD 8830M / R7 250 / R7 M465X] [1002:6... (rev 87)

When I add the video card gpu passthrough as a mac os , the proxmox server resets the whole machine.

My English is not very good, I tried to write with google translate. I hope I was able to explain my problem.

Thanks for your support.

REPLY



Nicholas Sherlock 

July 5, 2022 at 7:49 pm

Instead of using an SSDT you can just set the device ID to whatever you like in Proxmox. This way is much easier.

Do you get any crash logs printed to the host console? Or logged in /var/log/kern.log (or the previous .1 version of that file)

REPLY



Alper

July 6, 2022 at 6:48 am

How exactly should I do it in Proxmox? I need support on this matter.

At the moment, the card appears with the following ID.

05:00.0 VGA compatible controller [0300]: Advanced Micro Devices, Inc. [AMD/ATI] Cape Verde PRO / Venus LE / Tropo PRO-L [Radeon HD 8830M / R7 250 / R7 M465X] [1002:6... (rev 87)

How do I get it to look like this?

683dCape Verde XT [Radeon HD 7770/8760 / R7 250X]
3D680000

I think it is done from the place shown in the link here. but I don't quite understand how to do it.

<https://forum.proxmox.com/threads/pcie-passthrough-vendor-id.108935/>

I also reached the logs, but where exactly do I need to consider?
thnakyou for support.

REPLY



Nicholas Sherlock

July 6, 2022 at 7:10 am

You can put 0x1002 in the Vendor ID box and 0x683D in the Device ID box.

You want to look for logs that happened at the timestamp the VM crashed at

REPLY



Alper

July 6, 2022 at 7:53 am

When I enter the Vendor id and device id you have given, it boots without any problems and does not reset. However, the codes keep flowing in a loop on the Mac Loading screen. Mac os doesn't hurt. We did it right on the proxmox side, but I guess I need to do something on the mac os side?

But what can I do to make it hurt? Maybe it was Vendor id and Device id didn't like mac os Maybe I need to try another. However, with the Vendor id and Device id you provided, I was able to open in Ubuntu without any problems. Before entering these ids, the proxmox reset I wanted to do the Ubuntu gpu passthrough was happening. Thank you.

REPLY



alper

July 6, 2022 at 9:45 am

I was able to help by trying different ids here. 260X 280X 350X runs smoothly. However, the Graphics Card seems to be 3mb, how can I overcome this problem? My Graphics Card is actually 4GB.

REPLY



Nicholas Sherlock

July 6, 2022 at 9:46 am

That means you've picked an ID that macOS doesn't support acceleration for, so the AMD GPU driver isn't even being used. You'll need to pick another.

REPLY



Krishna

July 9, 2022 at 12:24 pm

Hey Is there any way to bypass RTX3050 Graphics card for hackintosh?

REPLY



Nicholas Sherlock

July 9, 2022 at 12:27 pm

No, absolutely none, the 2000 and 3000 series never had drivers written for them.

REPLY



Krishna

July 9, 2022 at 6:21 pm

Thanks for your response

REPLY



Krishna

July 9, 2022 at 6:21 pm

Hey, Is there any way to shutdown proxmox When hackintosh VM goes shutdown?

REPLY



Guedes

July 13, 2022 at 3:03 pm

Hi nick,
Thanks for your guide,

I've downloaded a full-img of Monterey using your script, but when i'm installing it on the vm i always get the error:

<https://imgur.com/uDJ0Xmm>

It seems that the installer cant validate the downloaded img.

Could yo please give some advice on what it's going on?

Thanks in advance!

Regards!

REPLY



Tomas

July 13, 2022 at 3:09 pm

Hi,
first, sorry fore noob question , but ... I installed OSX successfully, thanks for this tutorial. I can login to it via browser on another PC. But how can I setup physical PC, wher it is installed to boot OSX after start? When I start the PC, I still get PVE login

Thanks

REPLY



Nicholas Sherlock 

July 13, 2022 at 3:14 pm

To do that you would need to use PCIe passthrough to give a Mac compatible GPU to the guest

REPLY



Kartik Shenoy

July 14, 2022 at 5:14 pm

I was able to boot into MacOS. However, while re-reading through the post, I came across this:

> Note that you will want to hold off on logging into your Apple ID until you've configured your Mac's serial number in OpenCore (because otherwise a Mac with the default shared serial number in my OpenCore image will be added to your Apple ID).

I don't remember doing this. Could you please elaborate on how to replace the default shared serial with a serial from my MacBook?

REPLY



Nicholas Sherlock

July 14, 2022 at 8:10 pm

I wouldn't recommend cloning the serial from your MacBook. That'd look even odder for your account.

<https://dortania.github.io/OpenCore-Post-Install/universal/iservices.html#using-gensmbios>

REPLY



novra

July 15, 2022 at 7:07 pm

Hi, thank you for a great tutorial!

I am getting this error after I done the GPU passthrough:

kvm: -device vfio-pci,host=0000:28:00.0,id=hostpci0.0,bus=ich9-pcie-port-1,addr=0x0.0,multifunction=on: Failed to mmap 0000:28:00.0 BAR 0.
Performance may be slow

I already followed the guide on how to pass through the GPU. Here is my GPU:

8:00.0 VGA compatible controller [0300]: Advanced Micro Devices, Inc. [AMD/ATI] Navi 10 [Radeon RX 5600 OEM/5600 XT / 5700/5700 XT] [1002:731f] (rev ca)

It makes me stuck on the Apple logo while booting. Any idea how to solve this? Thank you!

REPLY



Nicholas Sherlock

July 15, 2022 at 9:28 pm

A host driver is still holding on to the card, so it can't be passed properly.

Most likely the culprit is bootfb, which you can fix by adding this to your host kernel args:

initcall_blacklist=sysfb_init

Note that your card suffers from the AMD Reset Bug and will only work once per host power cycle. Some cards can be fixed by adding "vendor-reset" to Proxmox

REPLY



novra

July 16, 2022 at 1:42 am

Hi, thank you for the assistance. I already added in the /etc/default/grub. So, it is now like this:

```
GRUB_DEFAULT=0
GRUB_TIMEOUT=5
GRUB_DISTRIBUTOR=`lsb_release -i -s 2> /dev/null || echo Debian`
GRUB_CMDLINE_LINUX_DEFAULT="quiet amd_iommu=on
iommu=pt pcie_acs_override=downstream,multifunction nofb
nomodeset video=vesafb:off,efifb:off,simplefb:off
initcall_blacklist=sysfb_init"
GRUB_CMDLINE_LINUX=""
```

I also followed your instruction to install vendor reset here <https://www.nicksherlock.com/2020/11/working-around-the-amd-gpu-reset-bug-on-proxmox/>

Finally, the error (Failed to mmap) is gone. I am not stuck in the apple logo (it is now showing progress bar loading). But, after the progress bar loading is gone, the screen turned off 😞 Did I do something wrong?

REPLY



Nicholas Sherlock

July 16, 2022 at 1:44 am

Most likely you still have the emulated video attached, and macOS is displaying there instead. Use vga: none in your VM config

REPLY



novra

July 16, 2022 at 2:08 am

Thank you Nick. Yup, I already added vga:none to my config. But my monitor still turned off after Apple logo loading finished.

REPLY



novra

July 16, 2022 at 4:09 am

ahh, I found the culprit. So I need to add agdpmod=pikera to the boot-args.

REPLY



TT

July 23, 2022 at 7:59 am

where did you add that?
I have the same issue. Monitor attached to 6600XT shows Proxmox logo, then opencore picker, then apple logo with loading bar for a few seconds, then no signal. VM is not crashed, still running, just no video output



Nicholas Sherlock

July 23, 2022 at 9:52 am

The boot-args are in OpenCore's config.plist:

<https://github.com/thenickdude/KVM-Opencore/blob/6750c11cd78696edd45a117753929c6999d551f2/EFI/OC/config.plist#L1138>

Instructions for editing that are in this blog post



S.

July 16, 2022 at 3:38 pm

Installing MacOS does not work anymore (for me) on Proxmox 7.2-7. I have just upgraded from 7.1-... to the latest version and now my installation is broken.

See also: <https://forum.proxmox.com/threads/macOS-monterey-not-booting-anymore-after-upgrade-to-7-2-7.112199/#post-484577>

Have you an idea how to come around this issue?

REPLY



Nicholas Sherlock

July 16, 2022 at 9:06 pm

Are you using GPU passthrough? Check host dmesg for errors

REPLY



S.

July 17, 2022 at 12:54 pm

No GPU passthrough. dmesg does not log any errors. I changes NUMA to 1 (as advised above) and now I get two different behaviors. In both cases the system stalls.

<https://wtf.roflcopter.fr/pics/gallery#r5CDMwrS/0HBYrE8N.png,4Q7U1yHy/m0NwePvF.png>

REPLY



S.

July 17, 2022 at 1:02 pm

My config: which was perfectly working with 7.1

```
acpi: 1
agent: 0
args: -device isa-applesmc,osk="..." -smbios type=2 -device usb-kbd,bus=ehci.0,port=2 -global nec-usb-xhci.msi=off -global ICH9-LPC.acpi-pci-hotplug-with-bridge-support=off -cpu Icelake-Server,kvm=on,vendor=GenuineIntel,+kvm_pv_unhalt,+kvm_pv_eoi,+hypervisor,+invtsc,+pcid,+ssse3,+sse4.2,+popcnt,+avx,+avx2,+aes,+fma,+fma4,+bmi1,+bmi2,+xsave,+xsaveopt,check
balloon: 0
bios: ovmf
boot: order=virtio0
cores: 4
cpu: Icelake-Server,flags=+aes
efidisk0: local-zfs:vm-110-disk-1,efitype=4m,size=1M
hotplug: disk,network,usb
kvm: 1
```


machine: q35
memory: 4096
name: b...
net0: virtio=...,bridge=vbr0,firewall=1
numa: 1
ostype: other
scsihw: virtio-scsi-pci
smbios1: uuid=60c2...
sockets: 1
startup: order=900
tablet: 1
vga: vmware
virtio0: local-zfs:vm-110-disk-
0,cache=unsafe,discard=on,size=128G
vmgenid: 2bf...

REPLY



S.

July 17, 2022 at 1:07 pm

Status running
HA State none
Node pve1
CPU usage 100.16% of 4 CPU(s)
Memory usage 13.93% (570.72 MiB of 4.00 GiB)
Bootdisk size 128.00 GiB

CPU permanently on 100% (moving between 99 and 101% – so system is doing something). memory usage not changing.

REPLY



S.

July 23, 2022 at 4:14 pm

Any news here?
With this configuration it is not even possible (for me) to install from scratch.
Installer disc image will not boot

REPLY



Brian Clarke

July 19, 2022 at 2:23 pm

Hey Nick, I've followed your guide, but hit a roadblock. When I start the newly created and edited VM, it does not load the open-core boot picker. I get a proxmox logo screen with the following errors.

```
BdsDxe: loading Boot0005 "UEFI QMU HARDDISK QM00003 " from
PciRoot (0x0) /Pci (0x1
F.0x2) /Sata (0x1.0xFFFF ,0x0)
BdsDxe: failed to load Boot0005 "UEFI QEMU HARDDISK QM00003 ",
from PciRoot (0x0) /
Pci (0x1F,0x2) /Sata (0x1,0xFFFF .0x0) : Access Denied
BdsDxe: failed to load Boot0006
"UEFI QEMU HARDDISK QM00001 " from PciRoot (0x0) /
Pci (0x1F,0x2) /Sata (0x0 ,0xFFFF •0x0) : Not Found
»»Start PXE over IP04.
```

When I run the "echo "options kvm ignore_msrs=Y" >>
/etc/modprobe.d/kvm.conf && update-initramfs -k all -u" command it returns:

update-initramfs: Generating /boot/initrd.img-5.13.19-2-pve
Running hook script 'zz-proxmox-boot'..
Re-executing '/etc/kernel/postinst.d/zz-proxmox-boot' in new private
mount namespace..
No /etc/kernel/proxmox-boot-uuids found, skipping ESP sync.

Any help or direction you could provide is greatly appreciated. Thank
you for documenting all of your hard work.

REPLY



Nicholas Sherlock 🇺🇸

July 19, 2022 at 5:34 pm

Remove the EFI disk from the Hardware page and delete it, then add
a replacement and this time be sure to un-tick the “pre-enroll keys”
tickbox

Also double check your boot order on the Options page

REPLY



TT

July 22, 2022 at 11:41 am

The solution for the “an error occurred..” error from the recovery image
is simple: don't use APFS! Use the old filesystem and it will work. (it did
at least in my OSX-KVM installation).

I have a much more basic error at my to following this guide.

I can't boot the opencore-v17.iso

I get access denied errors from the vm.

<https://posting.cc/pyCTfr1T>

REPLY



Nicholas Sherlock 🇺🇸

July 22, 2022 at 11:43 am

Remove the EFI disk from the Hardware page and delete it, then add
a replacement and this time be sure to un-tick the “pre-enroll keys”
tickbox

Also double check your boot order on the Options page

REPLY



TT

July 22, 2022 at 2:54 pm

oh no, i thought i did it right. Thanks a lot.

Your guide is amazing.

REPLY



TT

July 24, 2022 at 8:29 am

Thanks for all the help. but still no signal on monitor. “agdpmod=pikera”
did not help. I added the gpu rom file,..always the same.

I've installed TeamViewer inside of the vm to check if only the screen is black or if it freezes or something:

Well, macOS is working! it just does not detect the monitor, connected to one of the DisplayPort sockets at the graphics card!

<https://posting.cc/bD8Z6ZWw>

I also read in German it-news (heise.de) that macOS 12.5 is no longer working with OpenCore because apple changed something at the gpu handling. maybe this is related?

REPLY



Jonas

August 1, 2022 at 8:13 am

Thanks for the great tutorial.

The GPU passthrough is giving me problems at startup. I have a Firepro W7000.

On startup, the status bar under the Apple logo gets stuck at 1/3 and nothing happens.

If I take out the passthrough and go through the console, it starts normally.

However, the passthrough with the W7000 works fine with a Windows and a Linux VM on the same host.

It also works when installing MacOS. Until then the installation is finished and the "Main" operating system should start.

I also tested it with Big Sur – it makes no difference.

I do not get any error messages. Any idea what is I could still do to debug?

REPLY



Victor Hooi

August 5, 2022 at 4:54 pm

Hi Nick!

Do you happen to know if Mojave (macOS 10.14) would still work with these instructions, and using the current OpenCore (v17) image from the repo?

REPLY



Nicholas Sherlock

August 5, 2022 at 6:26 pm

I think that'll work fine, but with Mojave you need to use SATA disks instead of virtio-blk, and vmxnet3 networking instead of virtio-net.

REPLY

← OLDER COMMENTS

Leave a Reply

Your email address will not be published. Required fields are marked *

COMMENT *

NAME *

EMAIL *

WEBSITE

POST COMMENT

This site uses Akismet to reduce spam. [Learn how your comment data is processed.](#)

PREVIOUS

Recovering lost GPG public keys from your YubiKey

NEXT

Expanding the disk of your Proxmox macOS VM