# Ryzen and Threadripper(17h and 19h)

| Support | Version |
| --- | --- |
| Initial macOS Support | macOS 10.13, High Sierra |

## Starting Point

So making a config.plist may seem hard, it's not. It just takes some time but this guide will tell you how to configure everything, you won't be left in the cold. This also means if you have issues, review your config settings to make sure they're correct. Main things to note with OpenCore:

- **All properties must be defined**, there are no default OpenCore will fall back on so **do not delete sections unless told explicitly so**. If the guide doesn't mention the option, leave it at default.
- **The Sample.plist cannot be used As-Is**, you must configure it to your system
- **DO NOT USE CONFIGURATORS**, these rarely respect OpenCore's configuration and even some like Mackie's will add Clover properties and corrupt plists!

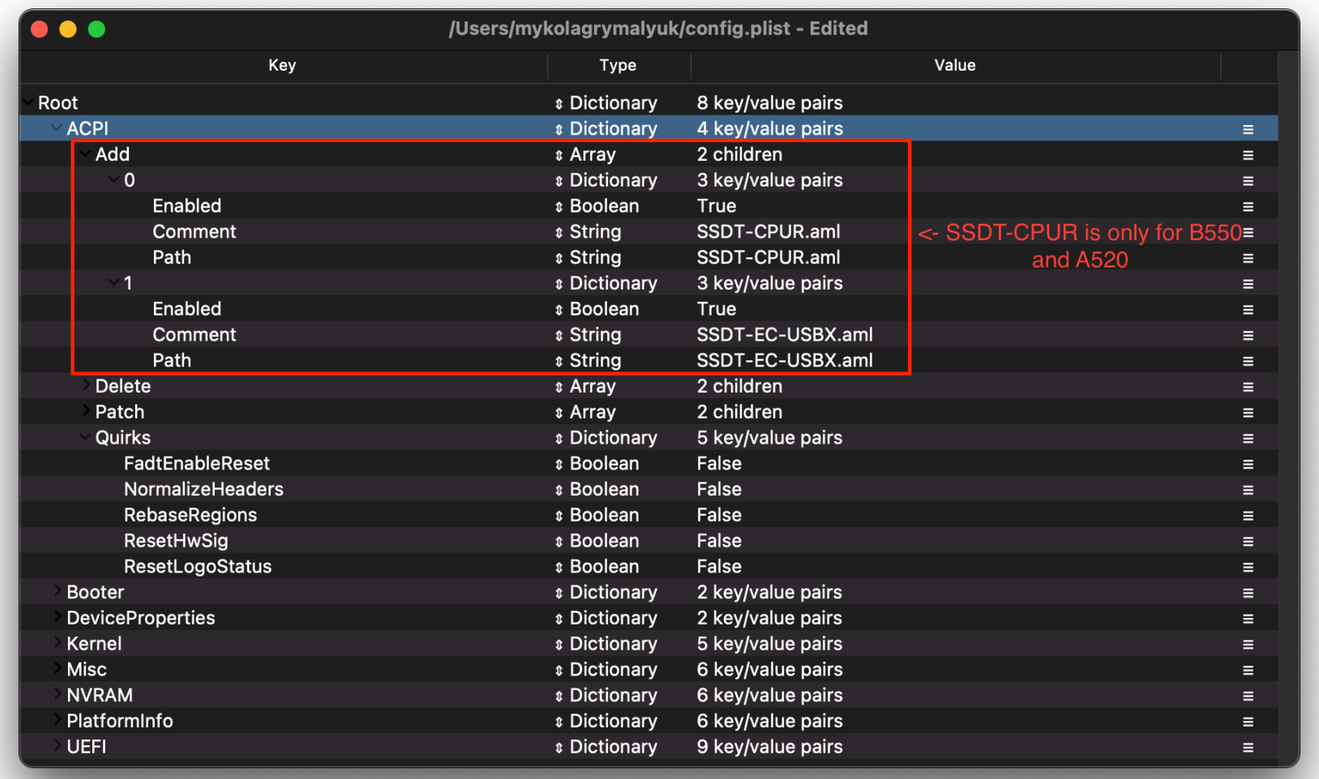
Now with all that, a quick reminder of the tools we need

- ProperTree
  - Universal plist editor
- GenSMBIOS
  - For generating our SMBIOS data
- Sample/config.plist
  - See previous section on how to obtain: config.plist Setup
- AMD Kernel Patches
  - Needed for booting macOS on AMD hardware(save these for later, we'll go over how to use them below)
  - Supporting AMD Family 15h, 16h, 17h and 19h

**WARNING**

Read this guide more than once before setting up OpenCore and make sure you have it set up correctly. Do note that images will not always be the most up-to-date so please read the text below them, if nothing's mentioned then leave as default.

## ACPI




### Add

**Info**

This is where you'll add SSDTs for your system, these are very important to **booting macOS** and have many uses like USB maps , disabling unsupported GPUs and such. And with our system, **it's even required to boot**. Guide on making them found here: **Getting started with ACPI**

| Required SSDTs | Description |
| --- | --- |
| SSDT-EC-USBX | Fixes both the embedded controller and USB power, see Getting Started With ACPI Guide for more details. |
| SSDT-CPUR | Fixes CPU definitions with B550 and A520 motherboards, **do not use** if you don't have an AMD B550 or A520 system. You can find a prebuilt here: SSDT-CPUR.aml |

Note that you **should not** add your generated `DSDT.aml` here, it is already in your firmware. So if present, remove the entry for it in your `config.plist` and under EFI/OC/ACPI.

`EFI/OC/ACPI` folder and **must** be specified in your config under `ACPI -> Add` as well.

### Delete

This blocks certain ACPI tables from loading, for us we can ignore this.

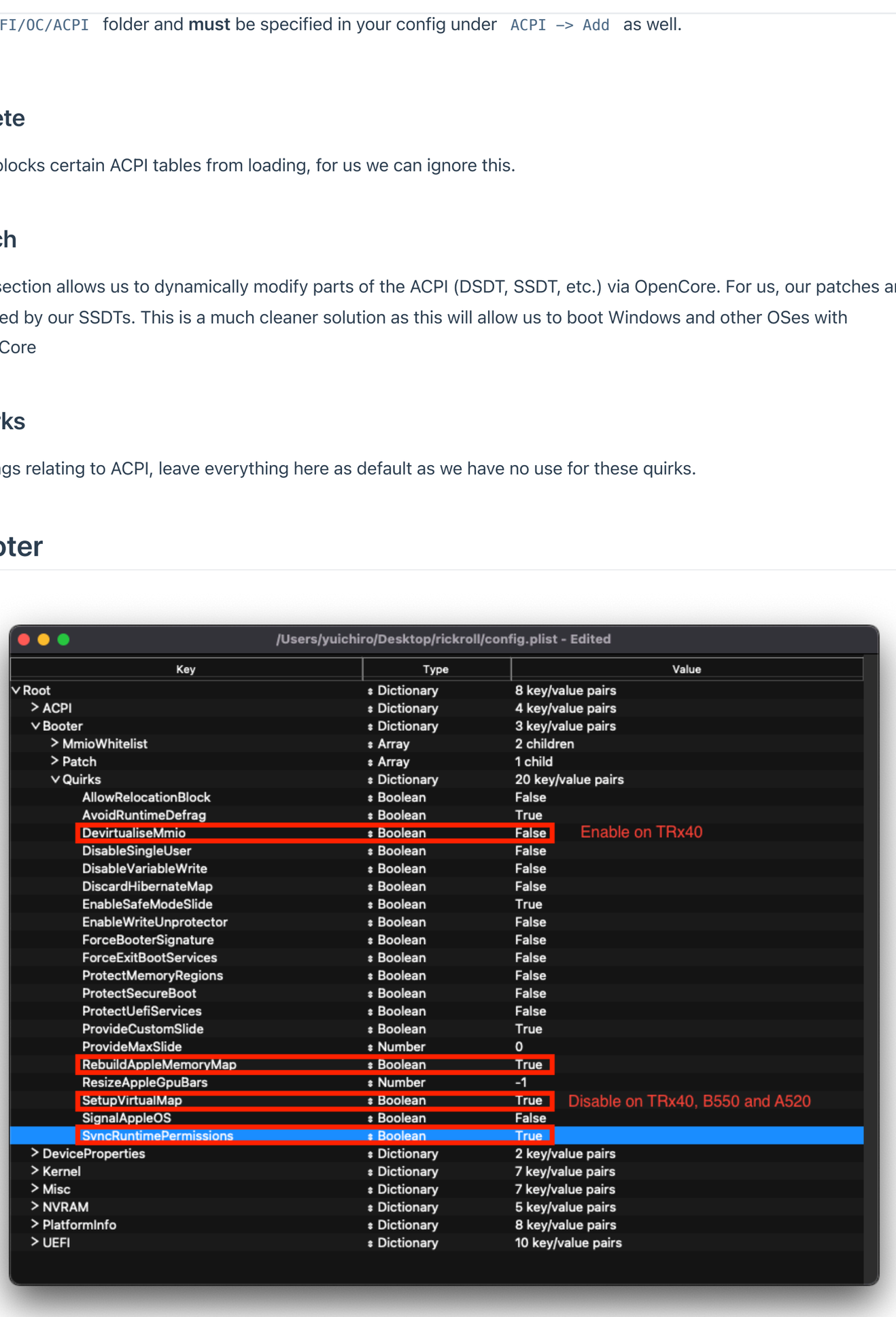
### Patch

This section allows us to dynamically modify parts of the ACPI (DSDT, SSDT, etc.) via OpenCore. For us, our patches are handled by our SSDTs. This is a much cleaner solution as this will allow us to boot Windows and other OSes with OpenCore

### Quirks

Settings relating to ACPI, leave everything here as default as we have no use for these quirks.

## Booter




This section is dedicated to quirks relating to boot.efi patching with OpenRuntime, the replacement for AptioMemoryFix.efi

### MmioWhitelist

This section is allowing spaces to be passthrough to macOS that are generally ignored, useful when paired with `DevirtualiseMmio`

### Quirks

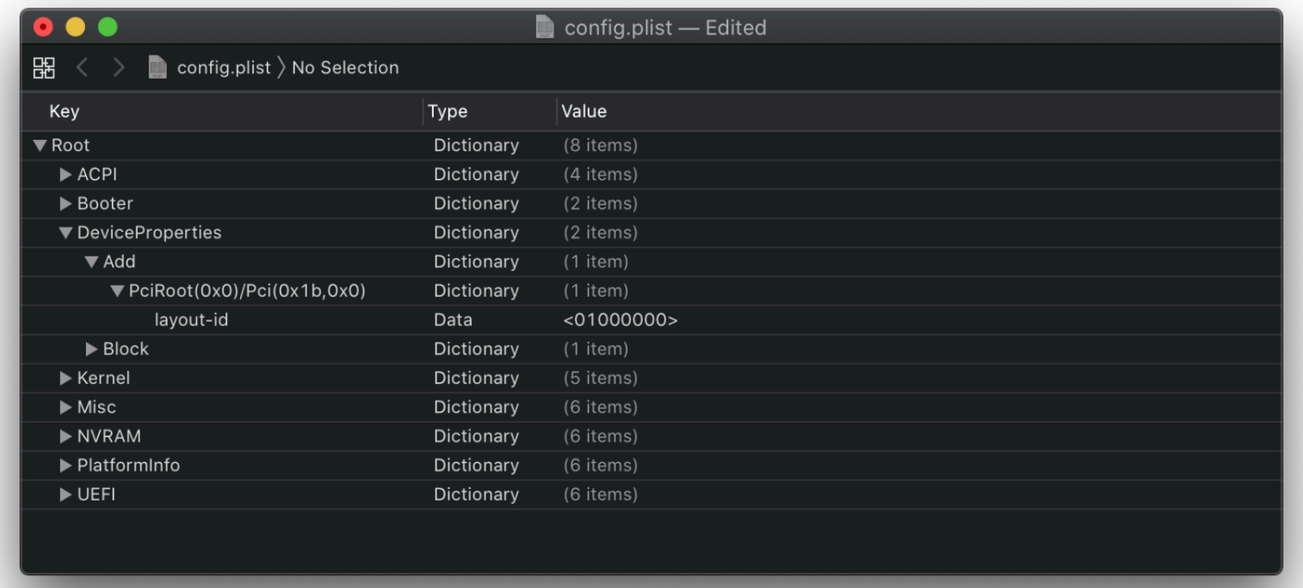
#### Info

Settings relating to boot.efi patching and firmware fixes, for us, we need to change the following:

| Quirk | Enabled | Comment |
|---|---|---|
| DevirtualiseMmio | NO | If you have a TRx40 system, enable this and follow the instructions here: https://dortania.github.io/OpenCore-Install-Guide/extras/kaslr-fix.html |
| EnableWriteUnprotector | NO | |
| RebuildAppleMemoryMap | YES | |
| ResizeAppleGpuBars | -1 | If your firmware supports increasing GPU Bar sizes (ie Resizable BAR Support), set this to `0` |
| SetupVirtualMap | YES | - Note X570, B550, A520 and TRx40 boards might need this disabled<br>- X470 and B450 with late 2020 BIOS updates might also require this disabled |
| SyncRuntimePermissions | YES | |

▶ More in-depth Info

# DeviceProperties




## Add

Sets device properties from a map.

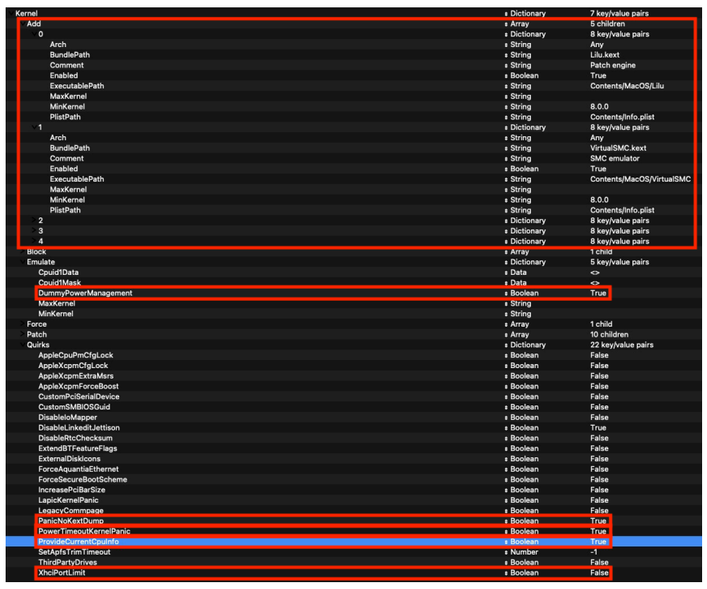
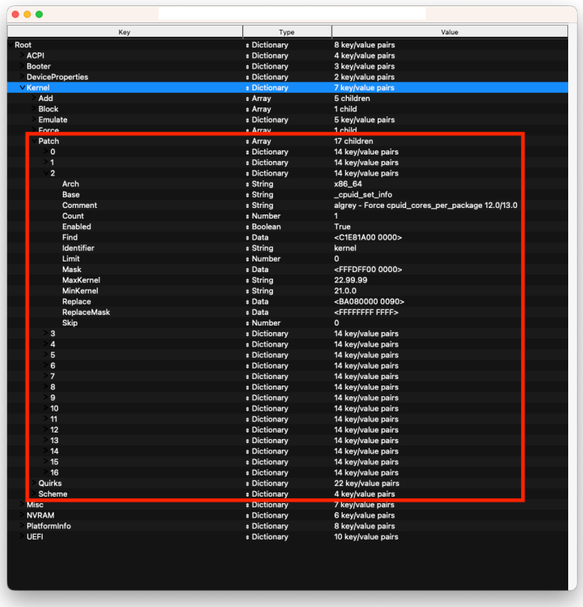
By default, the Sample.plist has this section set for audio which we'll be setting up by setting the layout ID in the boot-args section, so removal of `PciRoot(0x0)/Pci(0x1b,0x0)` is also recommended from the `Add` section.

TL;DR, delete all the PciRoot's here as we won't be using this section.

## Delete

Removes device properties from the map, for us we can ignore this

# Kernel

| Kernel | Kernel Patches |
| --- | --- |
|  |  |

## Add

Here's where we specify which kexts to load, in what specific order to load, and what architectures each kext is meant for. By default we recommend leaving what ProperTree has done, however for 32-bit CPUs please see below:

▶ More in-depth Info

## Emulate

**Info**

Needed for spoofing unsupported CPUs like Pentiums and Celerons and to disable CPU power management on unsupported CPUs (such as AMD CPUs)

| Quirk | Enabled |
| --- | --- |
| DummyPowerManagement | YES |

▶ More in-depth Info

## Force

For us, we can ignore.

## Block

Blocks certain kexts from loading. Not relevant for us.

## Patch

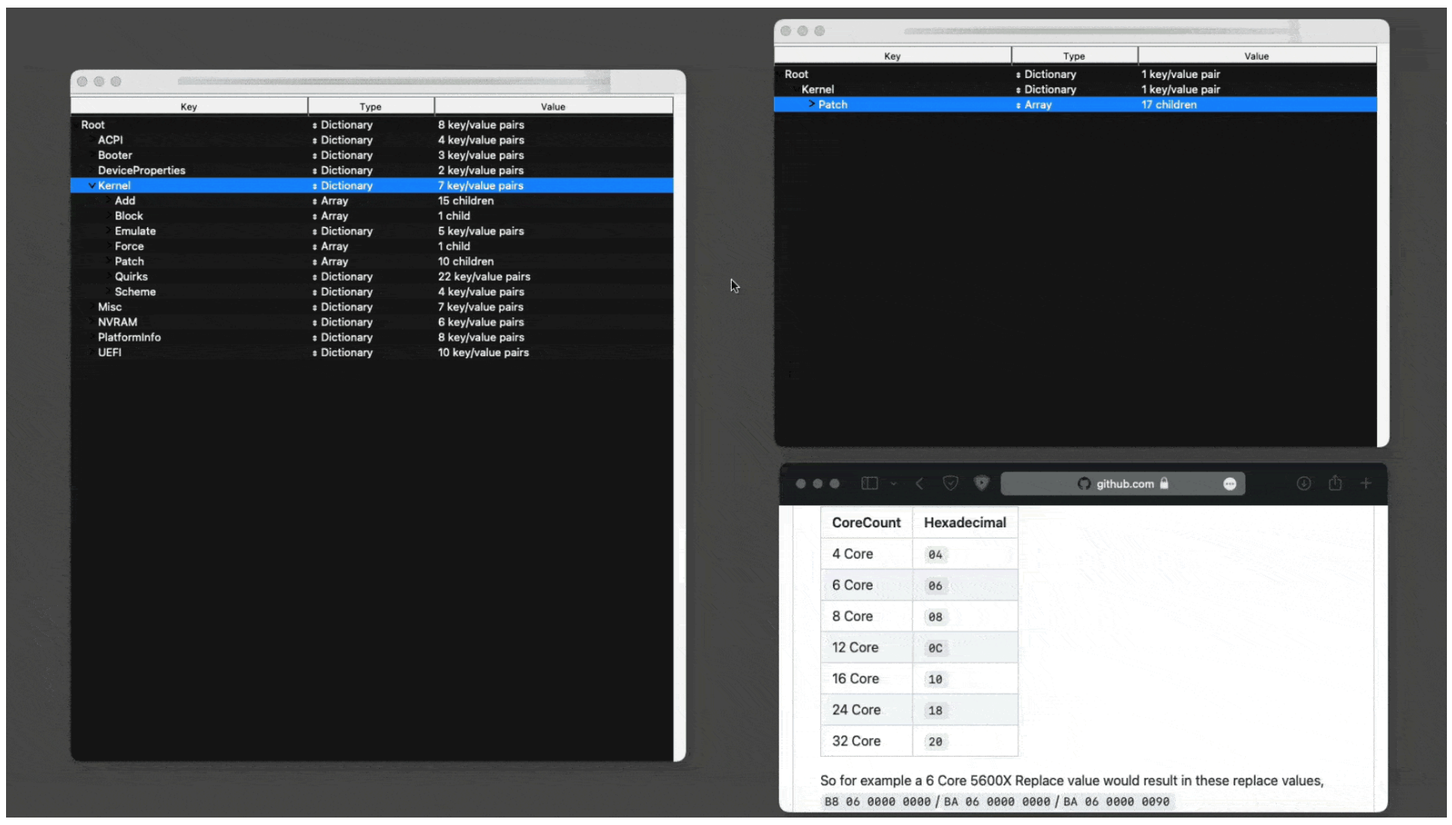
This is where the AMD kernel patching magic happens. Please do note that if coming from Clover, `KernelToPatch` and `MatchOS` from Clover becomes `Kernel` and `MinKernel` / `MaxKernel` in OpenCore. The latest AMD kernel patches can always be found on the AMD Vanilla GitHub Repository .

Kernel patches:

- Ryzen/Threadripper (17h/19h) (10.13 - 12.x)

To merge:

- Open both files,
- Delete the `Kernel -> Patch` section from config.plist
- Copy the `Kernel -> Patch` section from patches.plist
- Paste into where old patches were in config.plist




You will also need to modify four patches, all named `algrey - Force cpuid_cores_per_package` . You only need to change the `Replace` value. You should change:

- `B8000000 0000 => B8 <core count> 0000 0000`
- `BA000000 0000 => BA <core count> 0000 0000`
- `BA000000 0090 => BA <core count> 0000 0090`
- `BA000000 00 => BA <core count> 0000 00`

Where `<core count>` is replaced with the physical core count of your CPU in hexadecimal. For example, an 8-Core 5800X would have the new Replace value be:

- `B8 08 0000 0000`
- `BA 08 0000 0000`
- `BA 08 0000 0090`
- `BA 08 0000 00`

▶ Core Count => Hexadecimal Table

## Quirks

### Info

Settings relating to the kernel, for us we'll be enabling the following:

| Quirk | Enabled | Comment |
| --- | --- | --- |
| PanicNoKextDump | YES | |
| PowerTimeoutKernelPanic | YES | |
| ProvideCurrentCpuInfo | YES | |
| XhciPortLimit | YES | Disable if running macOS 11.3+ |

▶ More in-depth Info

## Scheme

▶ More in-depth Info

## Misc


### Boot

#### Info

| Quirk | Enabled | Comment |
| --- | --- | --- |
| HideAuxiliary | YES | Press space to show macOS recovery and other auxiliary entries |

▶ More in-depth Info

### Debug

#### Info

Helpful for debugging OpenCore boot issues:

| Quirk | Enabled |
| --- | --- |
| AppleDebug | YES |
| ApplePanic | YES |
| DisableWatchDog | YES |
| Target | 67 |

▶ More in-depth Info

### Security

Security is pretty self-explanatory, **do not skip**. We'll be changing the following:

| Quirk | Enabled | Comment |
|---|---|---|
| AllowSetDefault | YES | |
| BlacklistAppleUpdate | YES | |
| ScanPolicy | 0 | |
| SecureBootModel | Default | Leave this as `Default` for OpenCore to automatically set the correct value corresponding to your SMBIOS. The next page goes into more detail about this setting. |
| Vault | Optional | This is a word, it is not optional to omit this setting. You will regret it if you don't set it to Optional, note that it is case-sensitive |

▶ More in-depth Info

### Serial

Used for serial debugging (Leave everything as default).

### Tools

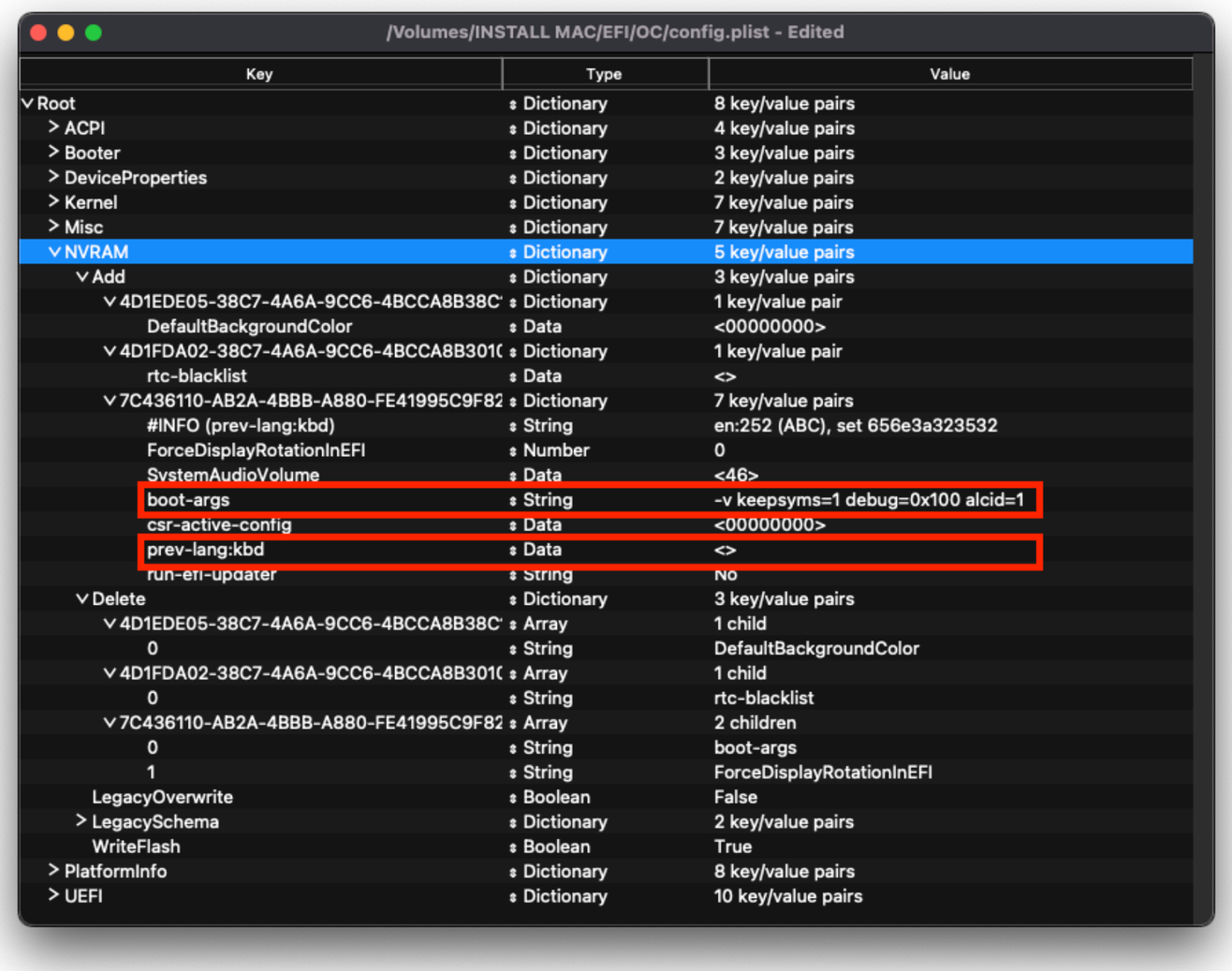
Used for running OC debugging tools like the shell, ProperTree's snapshot function will add these for you.

### Entries

Used for specifying irregular boot paths that can't be found naturally with OpenCore.

Won't be covered here, see 8.6 of Configuration.pdf for more info

## NVRAM




### Add

#### 4D1EDE05-38C7-4A6A-9CC6-4BCCA8B38C14

Used for OpenCore's UI scaling, default will work for us. See in-depth section for more info

▶ More in-depth Info

#### 4D1FDA02-38C7-4A6A-9CC6-4BCCA8B30102

OpenCore's NVRAM GUID, mainly relevant for RTCMemoryFixup users

▶ More in-depth Info

System Integrity Protection bitmask

- **General Purpose boot-args**:

| boot-args | Description |
|---|---|
| **-v** | This enables verbose mode, which shows all the behind-the-scenes text that scrolls by as you're booting instead of the Apple logo and progress bar. It's invaluable to any Hackintosher, as it gives you an inside look at the boot process, and can help you identify issues, problem kexts, etc. |
| **debug=0x100** | This disables macOS's watchdog which helps prevents a reboot on a kernel panic. That way you can *hopefully* glean some useful info and follow the breadcrumbs to get past the issues. |
| **keepsyms=1** | This is a companion setting to debug=0x100 that tells the OS to also print the symbols on a kernel panic. That can give some more helpful insight as to what's causing the panic itself. |
| **npci=0x3000** | This disables some PCI debugging related to `kIOPCIConfiguratorPFM64` and `gIOPCITunnelledKey`. This is an alternative to having Above 4G Decoding enabled in your BIOS. Do not use this unless you don't have it in your BIOS. Required for when getting stuck on `[PCI configuration begin]` as there are IRQ conflicts relating to your PCI lanes. Source |
| **alcid=1** | Used for setting layout-id for AppleALC, see supported codecs to figure out which layout to use for your specific system. More info on this is covered in the Post-Install Page |

- **GPU-Specific boot-args**:

| boot-args | Description |
|---|---|
| **agdpmod=pikera** | Used for disabling board ID checks on some Navi GPUs (RX 5000 & 6000 series), without this you'll get a black screen. **Don't use if you don't have Navi** (ie. Polaris and Vega cards shouldn't use this) |
| **-radcodec** | Used for allowing officially unsupported AMD GPUs (spoofed) to use the Hardware Video Encoder |
| **radpg=15** | Used for disabling some power-gating modes, helpful for properly initializing AMD Cape Verde based GPUs |
| **unfairgva=1** | Used for fixing hardware DRM support on supported AMD GPUs |
| **nvda_drv_vrl=1** | Used for enabling NVIDIA's Web Drivers on Maxwell and Pascal cards in macOS Sierra and High Sierra |

- **csr-active-config**: `00000000`
  - Settings for 'System Integrity Protection' (SIP).
  - csr-active-config by default is set to `00000000` which enables System Integrity Protection. You can choose a number of different values but overall we recommend keeping this enabled for best security practices. More info can be found in our troubleshooting page: Disabling SIP
- **run-efi-updater**: `No`
  - This is used to prevent Apple's firmware update packages from installing and breaking boot order; this is important as these firmware updates (meant for Macs) will not work.
- **prev-lang:kbd**: <>
  - Needed for non-latin keyboards in the format of `lang-COUNTRY:keyboard`, recommended to keep blank though you can specify it(**Default in Sample config is Russian**):
  - American: `en-US:0` ( `656e2d55533a30` in HEX)
  - Full list can be found in AppleKeyboardLayouts.txt
  - Hint: `prev-lang:kbd` can be changed into a String so you can input `en-US:0` directly instead of converting to HEX
  - Hint 2: `prev-lang:kbd` can be set to a blank variable (eg. `<>`) which will force the Language Picker to appear instead at first boot up.

| Key | Type | Value |
|---|---|---|
| prev-lang:kbd | String | en-US:0 |

## Delete

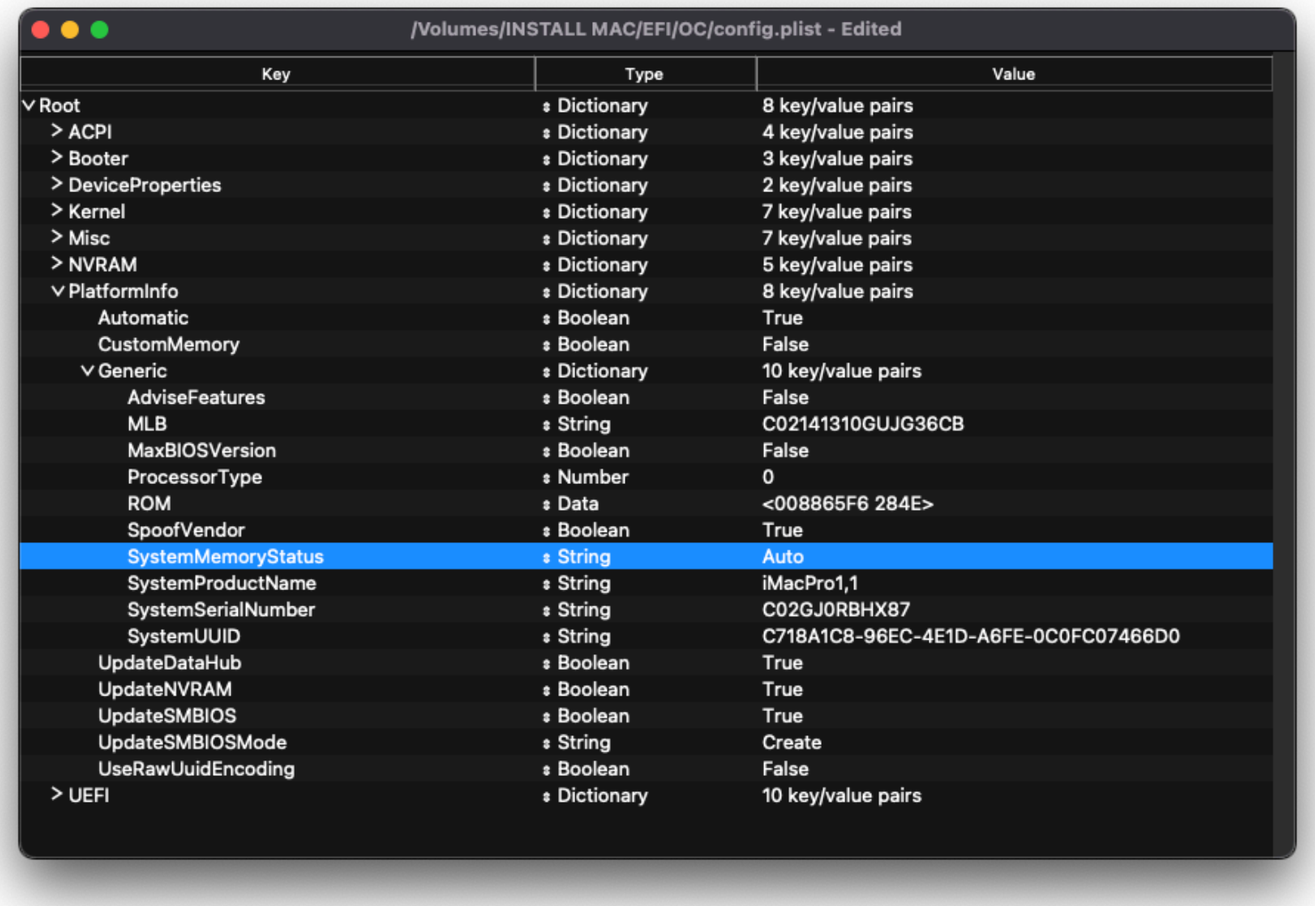
### Info

Forcibly rewrites NVRAM variables, do note that `Add` **will not overwrite** values already present in NVRAM so values like `boot-args` should be left alone. For us, we'll be changing the following:

| Quirk | Enabled |
|---|---|
| WriteFlash | YES |

▶ More in-depth Info

# PlatformInfo




### Info

For setting up the SMBIOS info, we'll use CorpNewt's GenSMBIOS application.

For this example, we'll choose the MacPro7,1 SMBIOS but some SMBIOS play with certain GPUs better than others:

- MacPro7,1: AMD Polaris and newer
  - Note that MacPro7,1 is exclusive to macOS 10.15, Catalina and newer
- iMacPro1,1: NVIDIA Maxwell and Pascal or AMD Polaris and newer
  - Use if you need High Sierra or Mojave, otherwise use MacPro7,1
- iMac14,2: NVIDIA Maxwell and Pascal
  - Use if you get black screens on iMacPro1,1 after installing Web Drivers with an NVIDIA GPU
- MacPro6,1: AMD GCN GPUs (supported HD and R5/R7/R9 series)

Run GenSMBIOS, pick option 1 for downloading MacSerial and Option 3 for selecting out SMBIOS. This will give us an output similar to the following:

```sh
  ##########################################################
 #              MacPro7,1 SMBIOS Info                     #
##########################################################

Type:         MacPro7,1
Serial:       F5KZV0JVP7QM
Board Serial: F5K9518024NK3F7JC
SmUUID:       535B897C-55F7-4D65-A8F4-40F4B96ED394
Apple ROM:    001D4F0D5E22
```

The order is `Product | Serial | Board Serial (MLB)`

The `Type` part gets copied to Generic -> SystemProductName.

The `Serial` part gets copied to Generic -> SystemSerialNumber.

The `Board Serial` part gets copied to Generic -> MLB.

The `SmUUID` part gets copied to Generic -> SystemUUID.

The `Apple ROM` part gets copied to Generic -> ROM.

**Reminder that you need an invalid serial! When inputting your serial number in Apple's Check Coverage Page , you should get a message such as "Unable to check coverage for this serial number."**

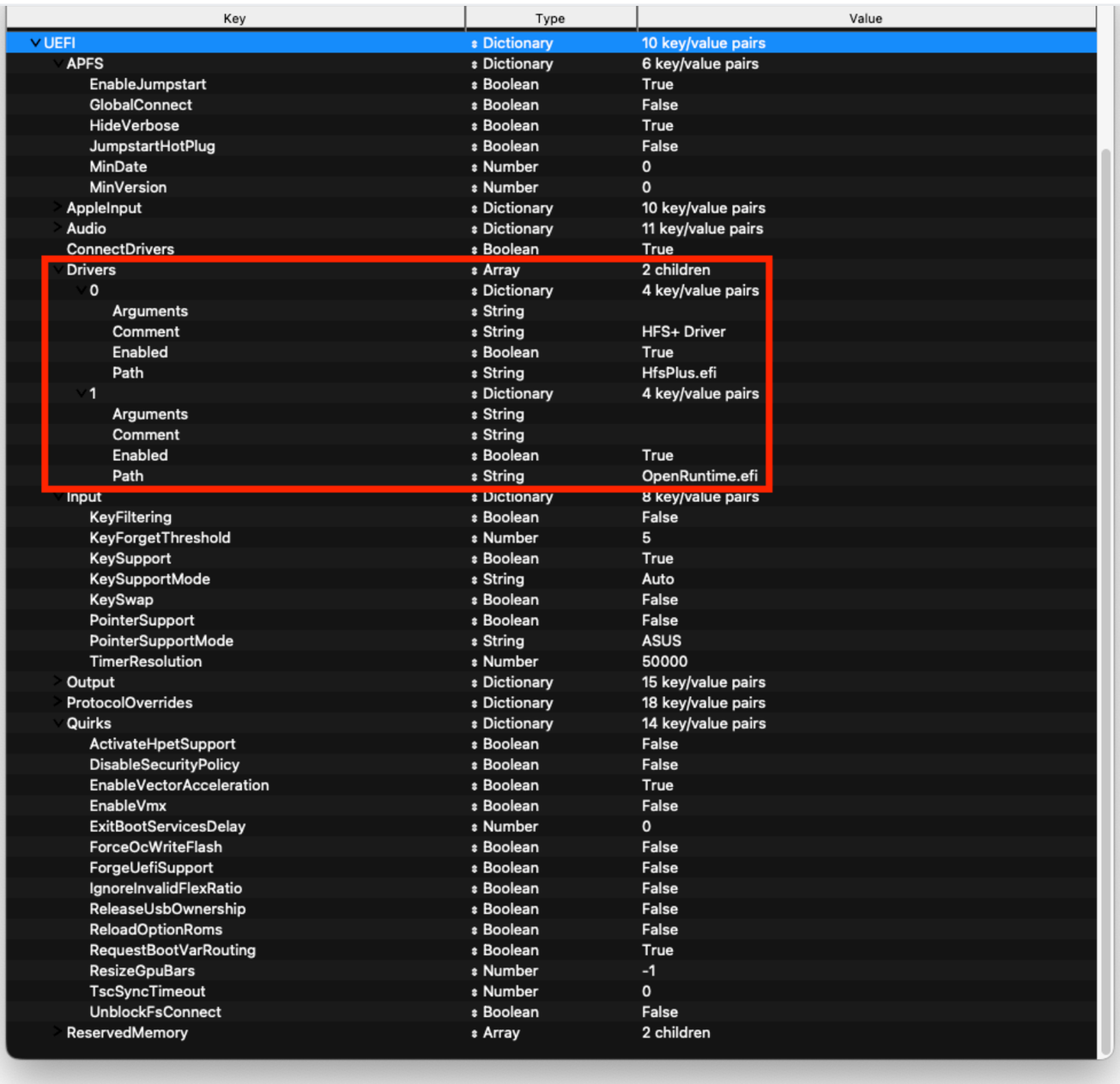
**Automatic**: YES

- Generates PlatformInfo based on Generic section instead of DataHub, NVRAM, and SMBIOS sections

## Generic

▶ More in-depth Info

# UEFI




**ConnectDrivers**: YES

- Forces .efi drivers, change to NO will automatically connect added UEFI drivers. This can make booting slightly faster, but not all drivers connect themselves. E.g. certain file system drivers may not load.

## Drivers

Add your .efi drivers here.

Only drivers present here should be:

- HfsPlus.efi
- OpenRuntime.efi

▶ More in-depth Info

## APFS

By default, OpenCore only loads APFS drivers from macOS Big Sur and newer. If you are booting macOS Catalina or earlier, you may need to set a new minimum version/date. Not setting this can result in OpenCore not finding your macOS partition!

macOS Sierra and earlier use HFS instead of APFS. You can skip this section if booting older versions of macOS.

### APFS Versions

Both MinVersion and MinDate need to be set if changing the minimum version.

| macOS Version | Min Version | Min Date |
|---|---|---|
| High Sierra ( `10.13.6` ) | `748077008000000` | `20180621` |
| Mojave ( `10.14.6` ) | `945275007000000` | `20190820` |
| Catalina ( `10.15.4` ) | `1412101001000000` | `20200306` |
| No restriction | `-1` | `-1` |

## Audio

Related to AudioDxe settings, for us we'll be ignoring(leave as default). This is unrelated to audio support in macOS.

- For further use of AudioDxe and the Audio section, please see the Post Install page: Add GUI and Boot-chime

## Input

Related to boot.efi keyboard passthrough used for FileVault and Hotkey support, leave everything here as default as we have no use for these quirks. See here for more details: Security and FileVault

## Output

▶ More in-depth Info

### ProtocolOverrides

Mainly relevant for Virtual machines, legacy macs and FileVault users. See here for more details: Security and FileVault

### Quirks

#### Info

Relating to quirks with the UEFI environment, for us we'll be changing the following:

| Quirk | Enabled | Comment |
|---|---|---|
| UnblockFsConnect | NO | Needed mainly by HP motherboards |

▶ More in-depth Info

### ReservedMemory

Used for exempting certain memory regions from OSes to use, mainly relevant for Sandy Bridge iGPUs or systems with faulty memory. Use of this quirk is not covered in this guide

## Cleaning up

And now you're ready to save and place it into your EFI under EFI/OC.

For those having booting issues, please make sure to read the Troubleshooting section first and if your questions are still unanswered we have plenty of resources at your disposal:

- AMD OS X Discord
- r/Hackintosh Subreddit

## AMD BIOS Settings

- Note: Most of these options may not be present in your firmware, we recommend matching up as closely as possible but don't be too concerned if many of these options are not available in your BIOS

### Disable

- Fast Boot
- Secure Boot
- Serial/COM Port
- Parallel Port
- Compatibility Support Module (CSM) (**Must be off in most cases, GPU errors/stalls like `gIO` are common when this option is enabled**)
- IOMMU

**Special note for 3990X users**: macOS currently does not support more than 64 threads in the kernel, and so will kernel panic if it sees more. The 3990X CPU has 128 threads total and so requires half of that disabled. We recommend disabling hyper threading in the BIOS for these situations.

### Enable

- Above 4G Decoding (**This must be on, if you can't find the option then add `npci=0x3000` to boot-args. Do not have both this option and npci enabled at the same time.**)
  - If you are on a Gigabyte/Aorus or an AsRock motherboard, enabling this option may break certain drivers(ie. Ethernet) and/or boot failures on other OSes, if it does happen then disable this option and opt for npci instead
  - 2020+ BIOS Notes: When enabling Above4G, Resizable BAR Support may become an available on some X570 and newer motherboards. Please ensure that Booter -> Quirks -> ResizeAppleGpuBars is set to `0` if this is enabled.
- EHCI/XHCI Hand-off
- OS type: Windows 8.1/10 UEFI Mode (some motherboards may require "Other OS" instead)
- SATA Mode: AHCI

## Once done here, we need to edit a couple extra values. Head to the Apple Secure Boot Page