Disabling GPU

So you need to hide your unsupported GPU? Well with OpenCore things are slightly different, specifically that we need to specify to which exact device we want to spoof. There are 3 ways we can do this:

- Boot Flag
  - Disables all GPUs except the iGPU
- DeviceProperties
  - Disables GPU on a per-slot basis
- SSDT
  - Disables GPU on a per-slot basis

CSM must be off in the BIOS for the spoofing to work correctly, especially on AMD CPU based systems.

Boot Flag

By far the simplest way, all you need to do is add the following boot-arg:

`-wegnoegpu`

Do note that this will disable all GPUs excluding the iGPU.

DeviceProperties Method

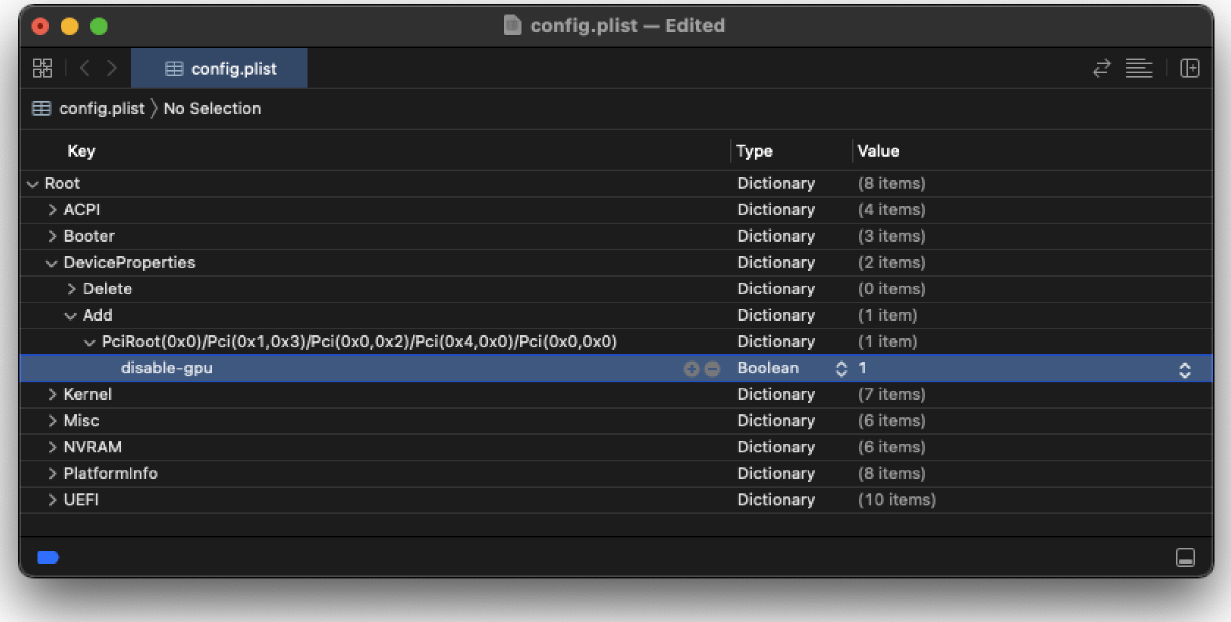
Here is quite simple, find the PCI route with gfxutil and then create a new DeviceProperties section with your spoof:

```
path/to/gfxutil -f GFX0
```

And the output will result in something similar:

With this, navigate towards `Root -> DeviceProperties -> Add` and add your PCI route with the following properties:

Key	Type	Value
disable-gpu	Boolean	`True`



Key	Type	Value
Root	Dictionary	(8 items)
ACPI	Dictionary	(4 items)
Booter	Dictionary	(3 items)
DeviceProperties	Dictionary	(2 items)
Delete	Dictionary	(0 items)
Add	Dictionary	(1 item)
PciRoot(0x0)/Pci(0x1,0x3)/Pci(0x0,0x2)/Pci(0x4,0x0)/Pci(0x0,0x0)	Dictionary	(1 item)
disable-gpu	Boolean	1
Kernel	Dictionary	(7 items)
Misc	Dictionary	(6 items)
NVRAM	Dictionary	(6 items)
PlatformInfo	Dictionary	(8 items)
UEFI	Dictionary	(10 items)

SSDT Method

There are many ways to find the path but generally, the easiest way is to get into Device Manager under windows and find the PCI path.

Example of device path for `\_SB.PCI0.PEG0.PEGP` :

```
DefinitionBlock ("", "SSDT", 2, "DRTNIA", "spoof", 0x00000000)
{
    External (_SB_.PCI0.PEG0.PEGP, DeviceObj)
```

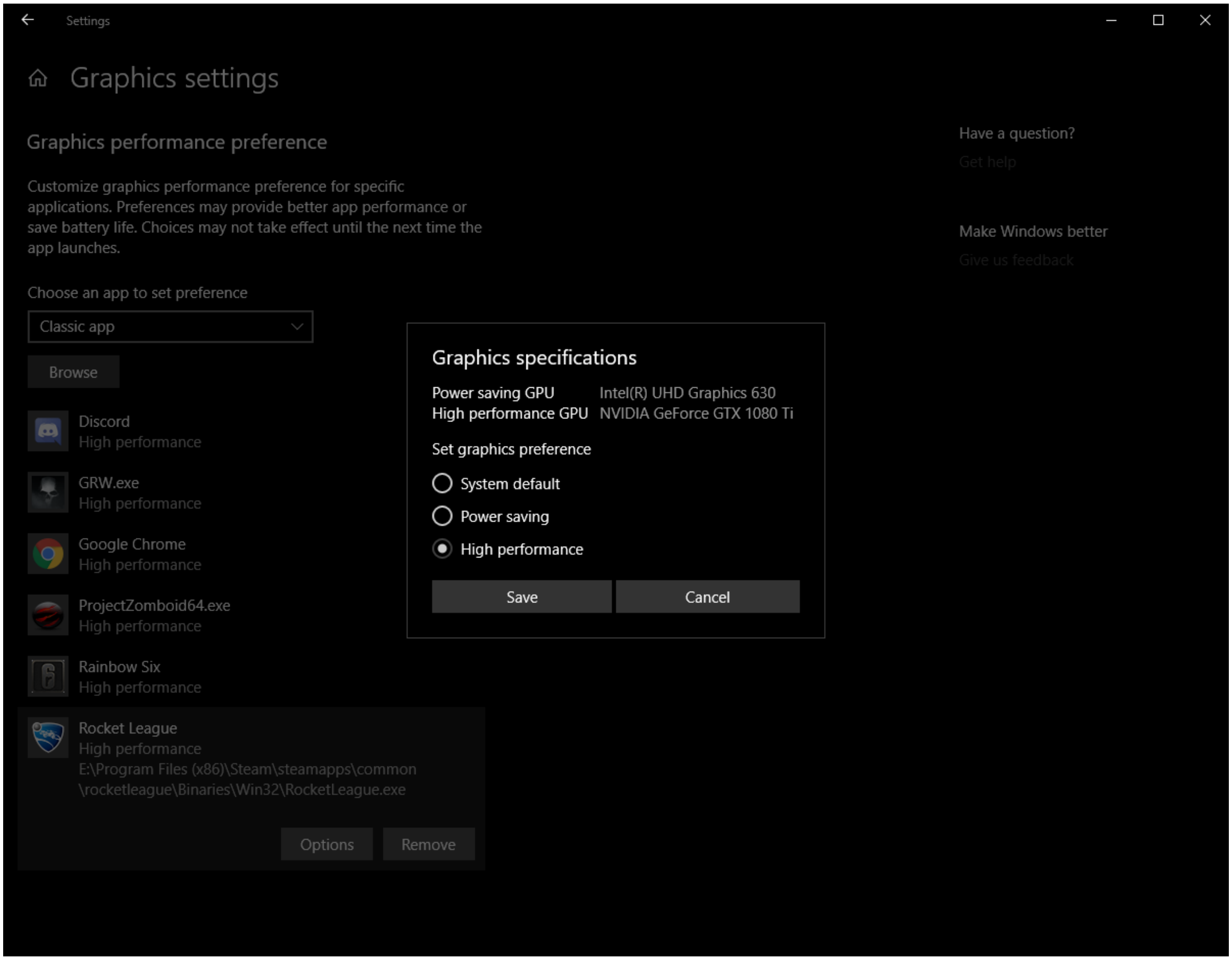
```
                If ((!Arg2 || !(_OSI ("Darwin"))))
                {
                    Return (Buffer (One)
                    {
                        0x03
                    })
                }

                Return (Package (0x0A)
                {
                    "name",
                    Buffer (0x09)
                    {
                        "#display"
                    },

                    "IOName",
                    "#display",
                    "class-code",
                    Buffer (0x04)
                    {
                        0xFF, 0xFF, 0xFF, 0xFF
                    },
                })
            }
        }
```

A copy of this SSDT can be found here: Spoof-SSDT.dsl . You will need MaciASL to compile this. Remember that `.aml` is assembled and `.dsl` is source code. You can compile with MaciASL by selecting File -> Save As -> ACPI Machine Language.

Source: CorpNewt

Windows GPU Selection

Many users only have two GPUs. NVIDIA and the Intel HD/UHD IGPU. Since NVIDIA no longer works on macOS, they may have the monitor plugged into the motherboards HDMI/DP connection for convenience. As a result, Windows will render all games and applications through the IGPU. You can reroute a specific game or application to a different GPU by going to: Settings > System > Display > Graphics settings



Graphics settings

Graphics performance preference

Customize graphics performance preference for specific applications. Preferences may provide better app performance or save battery life. Choices may not take effect until the next time the app launches.

Choose an app to set preference

Classic app

Discord
High performance

GRW.exe
High performance

Google Chrome
High performance

ProjectZomboid64.exe
High performance

Rainbow Six
High performance

Rocket League
High performance
E:\Program Files (x86)\Steam\steamapps\common\rocketleague\Binaries\Win32\RocketLeague.exe

Graphics specifications

Power saving GPU Intel(R) UHD Graphics 630
High performance GPU NVIDIA GeForce GTX 1080 Ti

Set graphics preference

○ System default
○ Power saving
● High performance

The rendered game or application will have its buffer copied to the IGPU. Which is then displayed to you. This does come with a few downsides:

- GSync will no longer work.
- NVIDIA settings can no longer be opened. This requires the display to be connected to the GPU
- Decreased frame rate.
- Increased input latency.

If your motherboard only has an HDMI connector for the IGPU, the maximum refresh rate for spec 2.1 is 120Hz . This assumes your board and monitor are of the same spec. This means your 144Hz monitor is only seeing a maximum of 120Hz as determined by the hardware. This limitation *does not* apply if your board has a DP connector for the IGPU.

If you have more than two GPUs (AMD, NVIDIA and Intel), this setting is limited. A monitor connected to the AMD GPU means Windows will only allow you to select the AMD GPU or the Intel IGPU. The NVIDIA GPU will not show. In a future version of Windows, this limitation is removed .

As a recommendation, if you use both operating systems equally and prefer no downsides, your best option is an HDMI or DP switch.

Help us improve this page!

Last Updated: 4/13/2023, 5:34:37 PM

← Fixing KASLR slide values

macOS 13: Ventura →