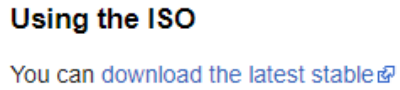
Proxmox GPU Passthrough into Windows 10

Please read carefully, I've mentioned everything if I'm not mistaken, but let me know if there is something wrong! Also ZOOM in if you need to view the images!!!

Guide to Follow: https://www.reddit.com/r/homelab/comments/b5xpua/the_ultimate_beginners_guide_to_gpu_passthrough/?utm_medium=android_app&utm_source=share

Required Files + Hardware:

- Proxmox Installation (ISO) - https://proxmox.com/en/downloads/category/iso-images-pve
- Windows 10 (ISO)
- VirtIO Drivers (ISO) - https://pve.proxmox.com/wiki/Windows_VirtIO_Drivers



- Hardware Capable of Virtualization
- Another Desktop/Laptop to Connect to Proxmox Web

BIOS Settings:

AMD

- Enable SVM
- Enable SR-IOV
- Enable Above 4G Decoding (If motherboard includes)
- Disable Re-size Bar

Intel

- Enable VT-D or Virtualization
- Enable SR-IOV
- Enable Above 4G Decoding (If motherboard includes)
- Disable Re-size Bar

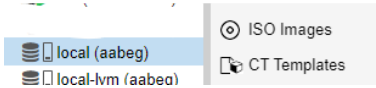
My Hardware:

- AMD Ryzen 5 3600X 6Core 12Thread
- ASUS X470 Pro
- RTX 3080TI
- 32GB Ram

How to install Proxmox if you haven't already

- > Go to https://proxmox.com/en/downloads/category/iso-images-pve
  > Install the latest ISO Image and Burn it onto a USB using Rufus > When you click run a pop-up will show > Select Write in DD-Image-Mode.

  Steps: *RUN WITHOUT { } – it is used to help you understand/read/copy it properly! :) | All commands are required to be run both on Intel & AMD*
- Upload your ISOs by going to Datacenter - Node Name - Local - ISO Images
  > Press Upload and select both your VirtIO and Win10 ISO files



- Edit Grub File
  > Access the Grub file using Shell Tab found on Proxmox Web
  > Run {nano /etc/default/grub}
  > REPLACE {GRUB_CMDLINE_LINUX_DEFAULT="quiet"} with
  {GRUB_CMDLINE_LINUX_DEFAULT="quiet intel_iommu=on"} -- For *Intel* CPU
  {GRUB_CMDLINE_LINUX_DEFAULT="quiet amd_iommu=on"} -- For *AMD* CPU
  {GRUB_CMDLINE_LINUX_DEFAULT="quiet amd_iommu=on iommu=pt pcie_acs_override=downstream,multifunction nofb nomodeset video=vesafb:off video=efifb:off"} -- *Advanced/Only if required* (replace {amd} with {intel} for Intel CPU - If your hardware is finicky
  > Once editing complete use <CTRL><S> to save and <CTRL><X> to exit
  > Run {update-grub}

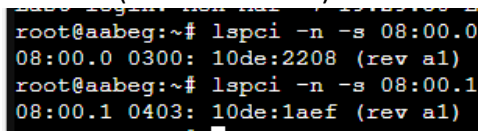
- Edit VFIO Modules
  > Access the Modules file using Shell Tab found on Proxmox Web
  > Run {nano /etc/modules}
  > Enter each line of following text on empty lines
  {vfio}
  {vfio_iommu_type1}
  {vfio_pci}
  {vfio_virqfd}
  > Once editing complete use <CTRL><S> to save and <CTRL><X> to exit

- IOMMU Interrupt Mapping
  > Paste the following text into Proxmox Shell {echo "options vfio_iommu_type1 allow_unsafe_interrupts=1" >

/etc/modprobe.d/iommu_unsafe_interrupts.conf}
> Paste the following text into Proxmox Shell {echo "options kvm ignore_msrs=1" > /etc/modprobe.d/kvm.conf}

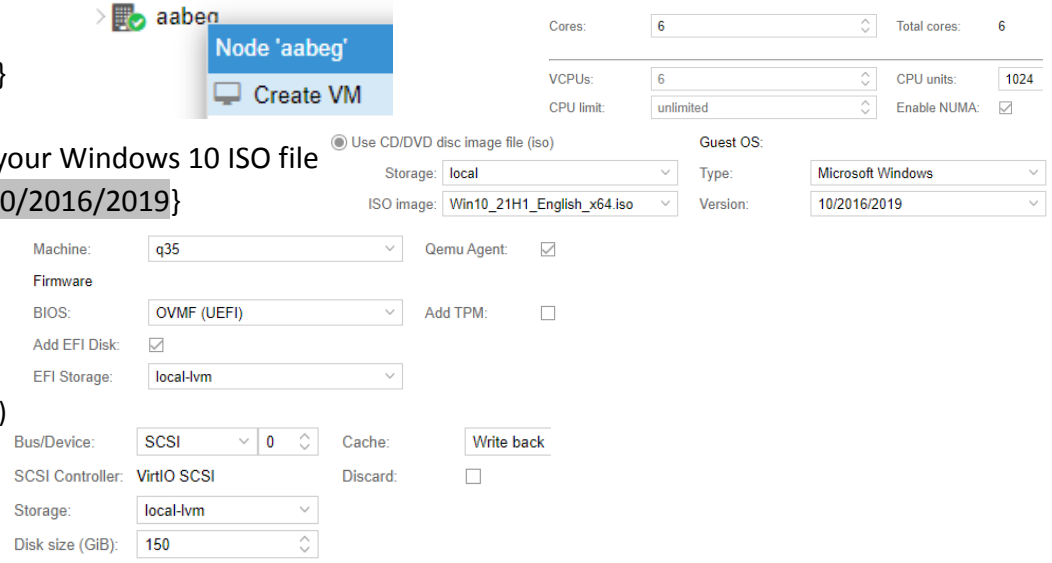
- Blacklist Drivers to Prevent Proxmox from using the GPUs
> Paste the following text into Proxmox Shell {echo "blacklist radeon" >> /etc/modprobe.d/blacklist.conf}
> Paste the following text into Proxmox Shell {echo "blacklist nouveau" >> /etc/modprobe.d/blacklist.conf}
> Paste the following text into Proxmox Shell {echo "blacklist nvidia" >> /etc/modprobe.d/blacklist.conf}

- Adding GPU to VFIO
> Paste the following text into Proxmox Shell {lspci -v}
> Scroll up until you find your desired GPU | For my GPU it's Identifier was 08:00.0 and 08:00.1 for the Audio (Both is needed)
> Paste the following text into Proxmox Shell with your OWN Identifier {lspci -n -s 08:00.0}
> Paste the following text into Proxmox Shell with your OWN Identifier {lspci -n -s 08:00.1}
> Read the output it would read similar to the image > > > >

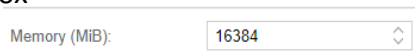


- > Paste the following text into Proxmox Shell with your OWN Identifier (the part where it says for me {10de:2208} and {10de:1aef})
{echo "options vfio-pci ids=10de:2208,10de:1aef disable_vga=1"> /etc/modprobe.d/vfio.conf}
> Paste the following text into Proxmox Shell {update-initramfs -u}
> Reboot your Proxmox Machine by running {reboot} into the Proxmox Shell

- Creating a New VM (HALFWAY THERE!) :)
> Right click on your Node Name and click on {Create VM}
> On the General Page - Add any name you wish!
> Go to the OS Page - From the dropdown menu click on your Windows 10 ISO file
> The for Type select {Microsoft Windows} and Version {10/2016/2019}
> Go to the System Page – Set Machine type to {Q35}
> Set BIOS type to {OVMF (UEFI)}
> Set EFI Storage to {local-lvm} or wherever desired
> Enable {Qemu Agent} make sure it's ticked
> - Optional - - Add TPM if you need Win11 (haven't tested yet)
> Go to Disks Page
> Set Bus/Device to {SCSI}
> Set Disk Size to your Liking I recommend at least 60Gigs
> Set Cache to {Write Back}
> Go to CPU Page

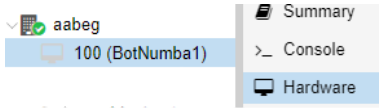


> Set Cores to your liking - Suggested - leave 1 core for the host - DO NOT COUNT THREADS ONLY CORES 6 core 12 thread = Set as 5/6 core
> Set {Enable Numa} to true or tick the box
> Go to Memory Page

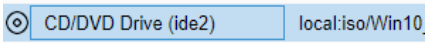


> Set Memory to your liking. 1024 = 1GB
> Disable {Ballooning Device}
> Go to Network Page
> Set Model to {VirtIO (paravirtualized)}
> Go to Confirm Page
> Click Finish but make sure {Start after created} is unticked or disabled

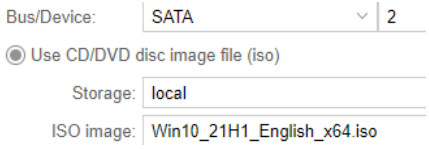
- Editing Hardware Options and Boot Menu (DON'T ADD GPU YET!)



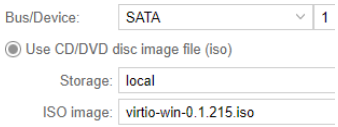
> Click on your newly created VM found below your Node Name
> Then go to the {Hardware} tab
> Click on {CD/DVD Drive (ide2)}



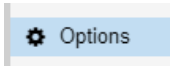
> Then click {Remove}
> Click Add - {CD/DVD Drive}
> Set Bus/Device to {SATA} with ID of {2}
> Set Storage to where you uploaded the ISO files – {local}
> Set ISO Image to {Win10} Iso file



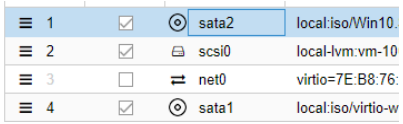
> Click Create, Now we will add VirtIO ISO
> Click Add - {CD/DVD Drive}
> Set Bus/Device to {SATA} with ID of {1}
> Set Storage to where you uploaded the ISO files – local
> Set ISO Image to VirtIO Iso file



> Now go to Options

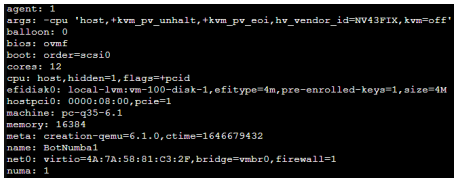


> Edit Boot Order
> Disable net0
> Drag sata2 (Win10ISO) to the Top
> Click Ok



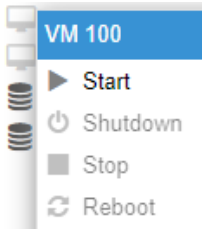
- Edit VM Config File

> Paste the following text into Proxmox Shell {nano /etc/pve/qemu-server/100.conf} - {100.conf} is according to your VM ID found before VM Name
> Enter each line of following text on empty lines (doesn't matter which line you paste, should be empty)

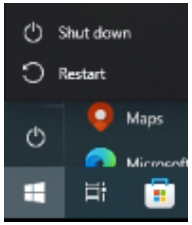


```
agent: 1
args: -cpu 'host,+kvm_pv_unhalt,+kvm_pv_eoi,hv_vendor_id=NV43FIX,kvm=off'
balloon: 0
bios: ovmf
boot: order=scsi0
cores: 12
cpu: host,hidden=1,flags=+pcid
efidisk0: local-lvm:vm-100-disk-1,efitype=4m,pre-enrolled-keys=1,size=4M
hostpci0: 0000:08:00,pcie=1
machine: pc-q35-6.1
memory: 16384
meta: creation-qemu=6.1.0,ctime=1646679432
name: BotNumba1
net0: virtio=4A:7A:58:81:C3:2F,bridge=vmbr0,firewall=1
numa: 1
```

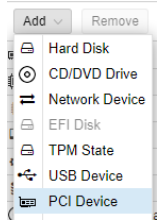
{cpu: host,hidden=1,flags=+pcid}
{args: -cpu 'host,+kvm_pv_unhalt,+kvm_pv_eoi,hv_vendor_id=NV43FIX,kvm=off'}
> Once editing complete use <CTRL><S> to save and <CTRL><X> to exit
> Reboot your Proxmox Machine by running {reboot} into the Proxmox Shell

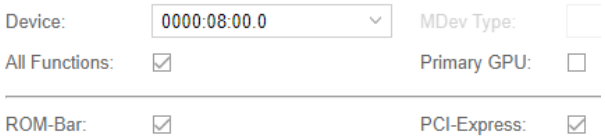


- Setup Windows/Install
> Right click and click on {Start} to boot
> When it mentions Press any Key to Boot from CD/DVD just spam your keyboard :)
If you see a different than the Windows setup, stop the VM and re-Start it
> Go through Windows Setup
> Once inside Windows, install {Guest Agent} by going into {File Explorer} > CD Drive of {VirtIO} > {guest-agent} > Run {qemu-ga-x86_64}
> Give Administrator Perms if requested
> Install Parsec or ANY remote desktop client (RDP)
> Do a test connection to the VM
> Once you're done shutdown the VM

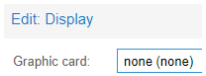


- Add the GPU (Finally! Last Step)
> Go back to the Hardware tab of your VM
> Click on Add
> {PCI Device}
> Click Device dropdown menu
> Locate your GPU (via it's ID ex. 08:00.0 – only the .0 not .1)
> Tick {Advanced}
> Tick {All Functions}, {ROM-Bar}, {PCI-Express}
> Click Add
> Should show similar to this >





> Boot the Machine and connect with your choice of RDP you should be welcomed with the home screen as it normally would
> Install GPU Drivers, NVIDIA or AMD
> Once you're all set turn the VM off
> Go to the Hardware Tab once again
> Set Display to {none (none)}



Congrats you set yourself up a VM with a GPU Passed into it. You are able to completely use the VM like you would with a normal machine

just with Proxmox on top!
Have fun! :)

Written by
- Aabeg :)

Install pve-headers:

Add "`deb http://download.proxmox.com/debian/pve buster pve-no-subscription`" to nano /etc/apt/sources.list

deb http://download.proxmox.com/debian/pve bullseye pvetest

Needed for Bluescreen avoidance in Windows AFAIK
nano /etc/modprobe.d/kvm.conf

```
#options kvm ignore_msrs=1
#options kvm report_ignored_msrs=0
```

https://www.reddit.com/r/Proxmox/comments/tyiorg/ryzen_gpupci_usb_passthrough_advanced/

https://forum.proxmox.com/threads/proxmox-pci-e-passthrough-to-host-monitor.89676/