

[GreenAsh](#)

01 Feb 2016

[Miscellaneous Computing](#)

# Running a real Windows install in VirtualBox on Linux

Having a complete Windows (or Mac) desktop running within Linux has been possible for some time now, thanks to the wonders of [Virtual Machine](#) (VM) technology. However, the typical approach is to mount and boot a VM image, where the guest OS and hard disk are just files on the host filesystem. In this case, the guest OS can't be natively booted and run, because it doesn't occupy its own disk or partition on the physical hardware, and therefore it can't be picked up by the BIOS / boot manager.

I've been installing Windows and Linux on the same machine, in a [dual-boot setup](#), for many years now. In this case, I boot natively into either one or the other of the installed OSes. However, I haven't run one "real" OS (i.e. an OS that's installed on a physical disk or partition) inside the other via a VM. At least, not until now.

At [my new job](#) this year, I discovered that it's possible to do such a thing, using a feature of [VirtualBox](#) called "[Raw Disk Access](#)". With surprisingly few hiccups, I got this running with [Linux Mint](#) 17.3 as the host, and with Windows 8.1 as the guest. Each OS is installed on a separate physical hard disk. I run Windows inside the VM most of the time, but I can still boot natively into the very same install of Windows at any time, if necessary.

## Instructions

This should go without saying, but please **back up** all your data before proceeding. What I'm explaining here is dangerous, and if anything goes wrong, you are likely to lose data on your PC.

If installing the two OSes on the same physical disk, then wipe the disk and [create partitions for each OS](#) as necessary (as is standard for dual-boot installs). (You can also shrink an existing Windows partition and then create the Linux partitions with the resulting free space, but this is more dangerous). If installing on different physical disks, then just keep reading.

Install Windows on its respective disk or partition (if it's not installed already, e.g. included with a home PC, SOE configured copy on a corporate PC). Windows should boot by default.

Go into your PC's BIOS setup (e.g. by pressing F12 when booting up), and ensure that "Secure Boot" and "Fast Boot" are disabled (if present), and ensure that "[Launch CSM](#)" / "[Launch PXE OpROM](#)" (or similar) are enabled (if present).

Install your preferred flavour of Linux on the other disk or partition. After doing this, GRUB should boot on startup, and it should let you choose to load Windows or Linux.

Install VirtualBox on Debian-based systems (e.g. Mint, Ubuntu) with:

```
sudo apt-get install virtualbox
```

```
sudo apt-get install virtualbox-dkms
```

Use a tool such as `fdisk` or `parted` to determine the partitions that the VM will need to access. In my case, for my Windows disk, it was partitions 1 (boot / EFI), 4 (recovery), and 5 (OS / "C drive").

Partition	File System	Mount Point	Label	Size	Used	Unused	Flags
/dev/sda1	fat32	/boot/efi	ESP	500.00 MiB	58.46 MiB	441.54 MiB	boot
/dev/sda2	fat32		DIAGS	40.00 MiB	4.00 MiB	36.00 MiB	hidden
/dev/sda3	unknown			128.00 MiB	---	---	msftres
/dev/sda4	ntfs		WINRETOOLS	750.00 MiB	293.12 MiB	456.88 MiB	hidden, diag
/dev/sda5	ntfs	/cdrive	OS	458.46 GiB	110.25 GiB	348.21 GiB	msftdata
/dev/sda6	ntfs		PBR Image	5.91 GiB	5.17 GiB	759.53 MiB	hidden, diag
unallocated	unallocated			1.00 MiB	---	---	

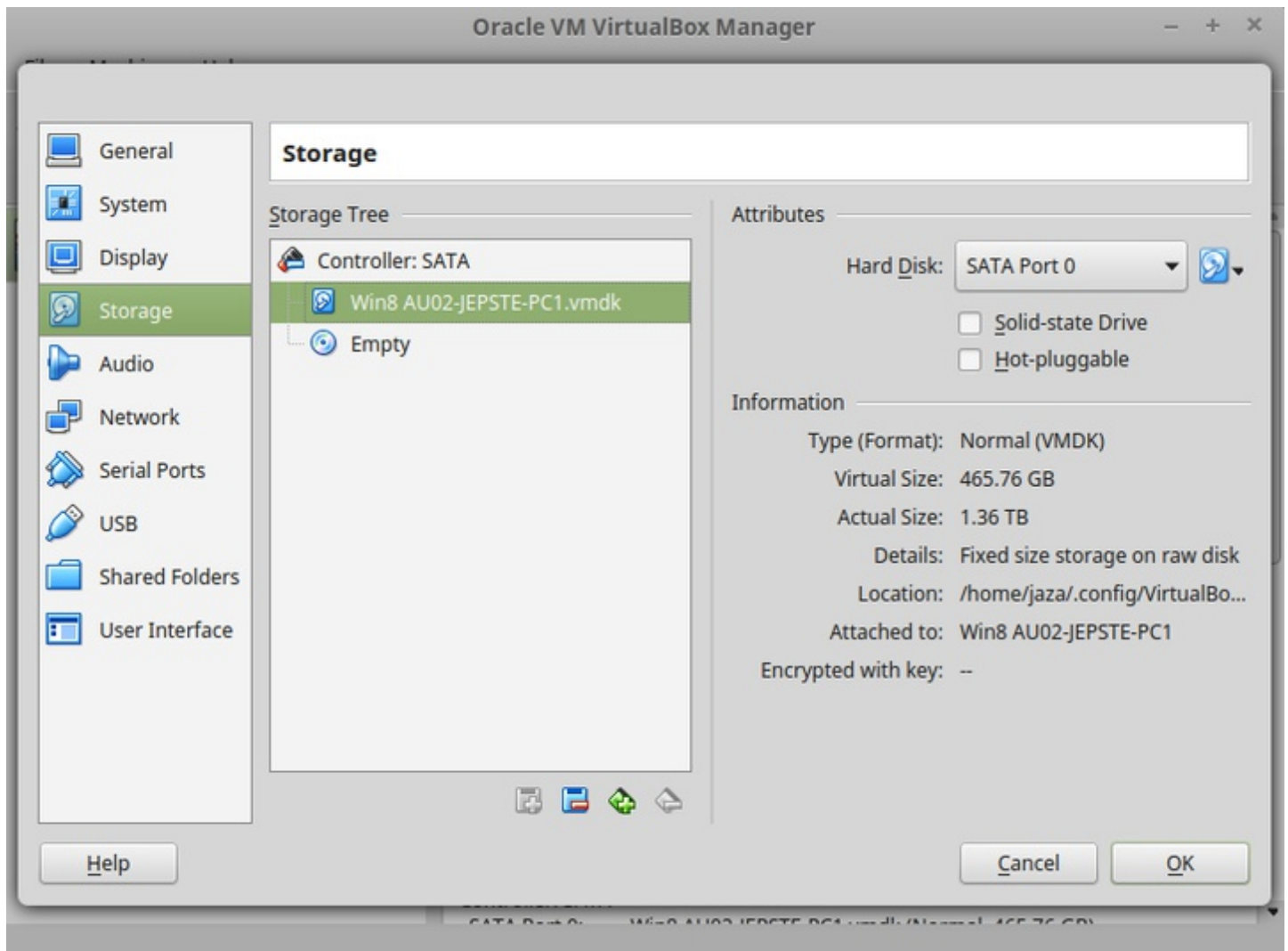
0 operations pending

*Partition table of my Windows disk as shown in GParted.*

Use this command (with your own filename / disk / partitions specified) to create the "raw disk", which is effectively a file that acts as a pointer to a disk / partition on which an OS is installed:

```
sudo VBoxManage internalcommands createrawvmdk \
-filename "/path/to/win8.vmdk" -rawdisk /dev/sda \
-partitions 1,4,5
```

Create a new VM in the VirtualBox GUI, with the OS and version that correspond to your install of Windows. In the "Storage" settings for the VM, add a hard disk (when prompted, click "Choose existing disk"), and point it to the .vmdk file that you created.



*VirtualBox treats the "raw" .vmdk file as if it were a virtual disk contained in a file.*

Start up your VM. You should see the same desktop that you have when you boot Windows natively!

Install VirtualBox Guest Additions as you would for a normal Windows VM, in order to get the usual VM bells and whistles (i.e. resizable window, mouse / clipboard integration, etc).

After you've been running your "real" Windows in the VM for a while, it will ask you to "Activate Windows". It will do this even if your Windows install is already activated when running natively. This is because Windows sees itself running within the VM, and sees "different hardware" (i.e. it thinks it's been installed on a second physical machine). You will have to activate Windows a second time within the VM (e.g. using a corporate bulk license key, by calling Microsoft, etc).

## Done

That's all there is to it. I should acknowledge that this guide is based on various [other guides](#) with similar instructions. Most online sources seem to very strongly warn that running Windows in this way is dangerous and can corrupt your system. Personally, I've now been running "raw" Windows in a VM like this every day for several weeks, with no major issues. The VM does crash sometimes (once every few days for me), as VMs do, and as Windows does. But nothing more serious than that.

I guess I should also warn readers of the potential dangers of this setup. It worked for me, but ~~YMMV~~. I've also heard rumour that on Windows 8 and higher, the problems of Windows not being able to adapt itself to boot on "different hardware" each startup (the real physical hardware, vs the hardware presented by VirtualBox) are much less than they used to be. Certainly doesn't seem to be an issue for me.

At any rate, I'm now happy; at least, as happy as someone who runs Windows in a VM all day can physically be. Hey, at least it's Linux outside that box on my screen. Good luck in having your cake and eating it, too.

**Comments are closed**

## Comment

27 Jul 2016

Leeo

[Permalink](#)

Thank you so much for sharing! That's worked for me :-D

ABN: 58 377 192 424

© 2004-2019 Jeremy Epstein. [Legalities](#) apply.