

Linux Virtuele Machinebeheerder

Ton Valkenburgh

In SoftwareBus 2022-1 heb ik de Linux virtuele machine Gnome-Boxes besproken. De instelmogelijkheden van Gnome-Boxes zijn echter beperkt. Met Virtuele Machinebeheerder is veel meer mogelijk. Daarmee is een virtuele machine te construeren die UEFI, Secure Boot en TPM ondersteunt. Hierop kan dus Windows 11 officieel worden geïnstalleerd.

Inleiding

Met de Virtuele Machinebeheerder kan een virtuele machine die met Gnome-Boxes ([link 1](#)) is aangemaakt worden aangepast. De met Gnome-Boxes aangemaakte machine heeft het oude BIOS. Helaas is het niet mogelijk om het BIOS naar UEFI om te zetten. Als we Windows 11 willen installeren moeten we daarom met de Virtuele Machinebeheerder een virtuele machine maken met de juiste opties. Als we deze machine met Gnome-Boxes willen gebruiken moeten we zorgen dat hij in de ruimte van Gnome-Boxes wordt aangemaakt. Met Virtuele Machinebeheerder maken we, net zoals bij Gnome-Boxes, gebruik van de machine-emulator QEMU ([link 2](#)). In de lijst van door Windows 11 ondersteunde processors komen de QEMU-machines niet voor. Toch blijkt het mogelijk om in de virtuele machine, zonder bypass-opties, Windows 11 te installeren.

Ik heb alle tests uitgevoerd onder Kubuntu 20.04 met als processor een AMD Ryzen 7 3700X en een Intel i7 6700HQ. Beide ondersteunen Secure Boot en de Trusted Platform Module (TPM), maar de Intel-processor wordt niet officieel ondersteund. In beide gevallen is Windows 11 volgens de standaard installatieprocedure te installeren in een virtuele machine.

Trusted Platform Module

Als de processor TPM ondersteunt kan deze worden gebruikt. Als je dat wilt, zorg je dat UEFI en TPM 2.0 zijn geactiveerd in het BIOS. De manier waarop dat moet gebeuren is afhankelijk van dat BIOS. Met wat zoeken in dat BIOS is dat wel te vinden. In de virtuele machine kunnen we dan gebruik maken van de TPM van de host. Het nadeel van deze methode is dat slechts één gast tegelijkertijd gebruik kan maken van de TPM. Bij een geëmuleerde TPM in de virtuele machine krijgt iedere gastmachine zijn eigen TPM. We kiezen daarom voor deze aanpak. De geëmuleerde TPM is niet in de repository (distributie) van Ubuntu 20.04 aanwezig. Daarom voegen we een extra repository toe en installeren dan de software TPM ([link 3](#)). Tik in de terminal:

```
sudo apt-add-repository ppa:smoser/swtpm
sudo apt install swtpm swtpm-tools
```

Nu moeten we nog een bestand toegankelijk maken voor je account met het commando:

```
sudo chown <je_account> /var/lib/swtpm-localca
```

Voor <je_account> vul je je eigen accountnaam in.

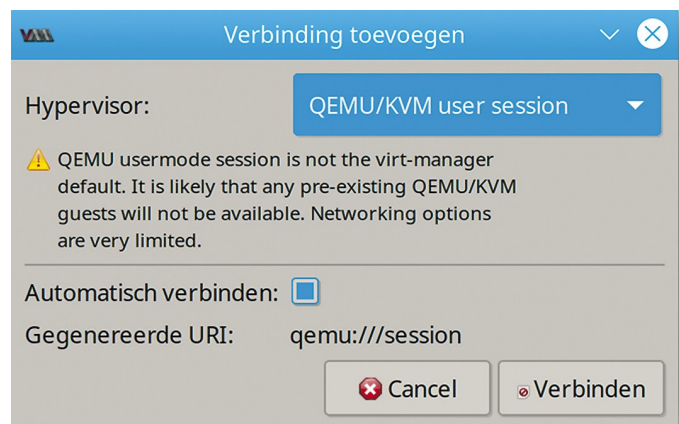
We gaan nu de Virtuele Machinebeheerder installeren

De Virtuele Machinebeheerder

Ik ga ervan uit dat Gnome-Boxes is geïnstalleerd zoals besproken in het artikel Gnome-Boxes in SoftwareBus 2022-1 ([link 4](#)). We gebruiken de Virtuele Machinebeheerder om de virtuele machine te configureren. Deze machine gebruiken we later met Gnome-Boxes.

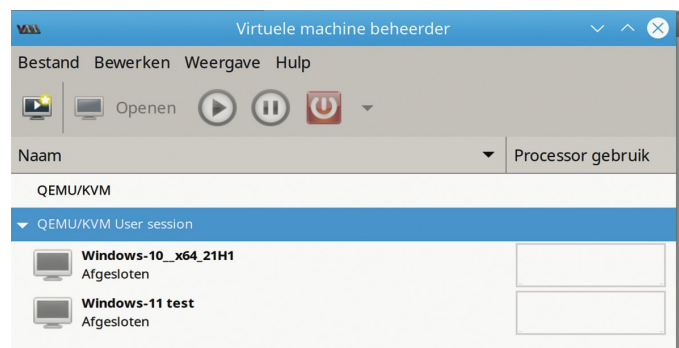
Installeer Virtuele Machinebeheerder (virt-manager) vanuit de distributie. Start de Virtuele Machinebeheerder. Standaard maakt de virt-manager verbinding met `qemu:///system`

Wij willen de machine echter in Gnome-Boxes kunnen gebruiken. Om dat te doen selecteer je *Bestand > Verbinding toevoegen*. In het nieuwe venster selecteer je met de pull-down: *QEMU/KVM user session* en klik je op *Verbinden* (zie afbeelding 1). QEMU/KVM user session is de ruimte die door Gnome-Boxes wordt gebruikt.



Afbeelding 1: Verbinding toevoegen

Je ziet nu de virtuele machines die ik heb aangemaakt met Gnome-Boxes (afbeelding 2).



Afbeelding 2: QEMU/KVM User session

We gaan de gewenste machine aanmaken. Klik met de rechtermuisknop op *QEMU/KVM Users Session*. In het pop-upvenster kies je *New*. In het nieuwe venster gebruiken we de defaults en klikken op *Forward*. Met de knop *Bladeren* selecteer je een image-bestand van Windows 11. Door virt.manager wordt dat als Windows versie 10 herkend. Kies *Forward*. Nu stellen we de grootte van het geheugen in op 8.000 MiB en het aantal CPU's op het maximum; in mijn geval voor de Ryzen: 16. Weer klikken we op *Forward*. We kiezen 128 GiB voor de schijf. Klik op *Forward*. Nu gaan we de machine aan-

passen. Verander de naam in bijvoorbeeld **Windows_11**. Selecteer *Pas configuratie aan voor installatie*. Klik op *Voltoeien*. Met het venster (afbeelding 3) dat verschijnt gaan we de machine bouwen.



Afbeelding 3: Machine details

je lokale account in te vullen. Vervolg de installatie.

Na installatie sluiten we Windows af. Daarna kiezen we *Weergave* en selecteren *Details*. We maken nu bij SATA CDROM 1 de CD ROM leeg. Kies *Apply*.

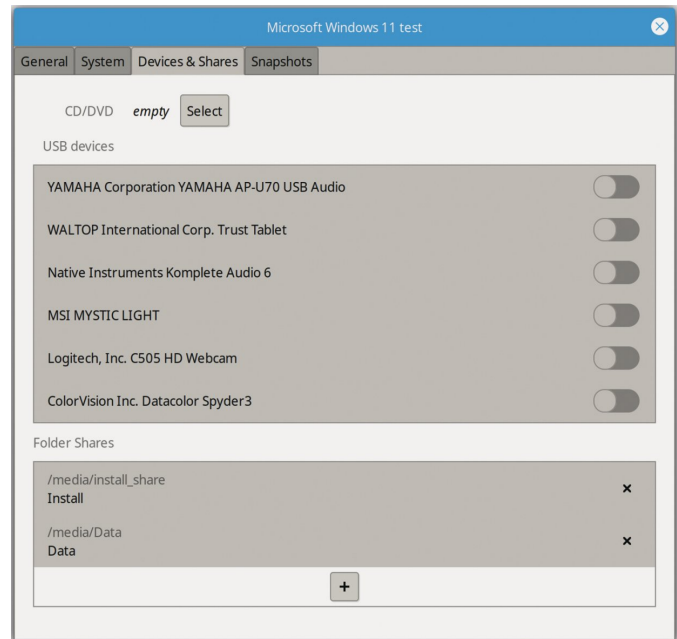
Sluit de Virtuele Machinebeheerder af en start Windows 11 weer op, maar nu vanuit Gnome-Boxes.

Gnome-Boxes

Bij een niet-geactiveerde Windows zijn instellingen beperkt te wijzigen. Dat geldt ook voor de schermresolutie. Dit laatste kunnen we oplossen door het installeren van een programma. Op de website van Spice-space ([link 5](#)) halen we de *Windows guest tools* (spice-guest-tools) op. Na installatie is de beperking in de schermresolutie vervallen. De resolutie past zich nu automatisch aan naar de venstergrootte. De muis kan zich nu soepel bewegen van gast naar host en omgekeerd. Verder kun je nu teksten en afbeeldingen kopiëren en plakken tussen host en gast. Om bestanden te kunnen delen halen we de *Spice WebDAV daemon* vanaf dezelfde website op en installeren deze. Na installatie en herstarten kunnen we een netwerkstation aanmaken met toegang tot mappen van de host. Ga in Gnome-Boxes naar de bovenste balk rechts, klik op de drie puntjes en selecteer *Properties*. In het verschenen venster kies je *Devices & Shares*. Hier kun je kiezen welke USB-apparaten je wilt gebruiken en door op de *plus* te klikken kun je mappen van de host toevoegen. Afbeelding 4 toont de door mij gekoppelde NAS-share:

install_share en de hostmap Data.

Door middel van functies onder de andere tabs kun je diverse aanpassingen maken en snapshots opslaan.



Afbeelding 4: Devices & Shares

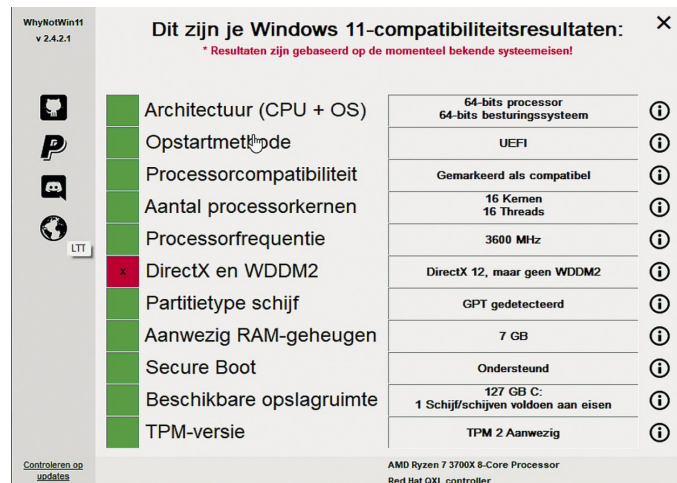
Het delen van bestanden met de Spice WebDAV daemon gaat prima met kleine bestanden. Wil je grotere bestanden delen, dan zul je een wijziging in het Windows register moeten aanbrengen. Open het register en ga naar:

`HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\WebClient\Parameters`. Dubbelklik op `FileSizeLimitInBytes`. Vul daar de maximumwaarde van 4 GB hexadecimaal in als `fffffff`. Als de sleutel niet aanwezig is maak je de sleutel `WebClientParameters` aan en daarin het `DWORD` (32 bits)-waarde `FileSizeLimitInBytes` met achtmaal `f`.

Je kunt in de Virtuele Machinebeheerder de configuratie altijd aanpassen. De uitzondering is de Chipset Q35 en de Firmware UEFI. Verder is het mij opgevallen dat de test op comptabiliteit bij het installeren van Windows 11 niet altijd volledig is.

Zo blijkt WDDM2 niet nodig te zijn (zie afbeelding 5).

Er blijkt echter op meer items niet te worden getest. Na de installatie van Windows 11 op een Intel i7 HQ6700 krijgen we het resultaat van afbeelding 6.



Afbeelding 5: Why Not Windows 11

Ook op de processor-comptabiliteit wordt bij de installatie in de virtuele machine niet getest. Windows 11 werkt echter prima. Of dat zo blijft, zou ik niet weten. Ik heb geconstateerd dat de test in ieder geval op UEFI, aantal kernen, grootte van de schijf, Secure Boot en TPM plaatsvindt.

WhyNotWin11 v.2.4.2.1

Dit zijn je Windows 11-compatibiliteitsresultaten: ✕

* Resultaten zijn gebaseerd op de momenteel bekende systeemeisen!

Architectuur (CPU + OS)	64-bits processor 64-bits besturingssysteem	i
Opstartmethode	UEFI	i
Processorcompatibiliteit	Niet-ondersteund	i
Aantal processorkernen	4 Kernen 8 Threads	i
Processorfrequentie	2600 MHz	i
DirectX en WDDM2	DirectX 12, maar geen WDDM2	i
Partitietype schijf	GPT gedetecteerd	i
Aanwezig RAM-geheugen	7 GB	i
Secure Boot	Ondersteund	i
Beschikbare opslagruimte	127 GB C: 1 Schijf/schijven voldoen aan eisen	i
TPM-versie	TPM 2 Aanwezig	i

Controleren op updates

Intel(R) Core(TM) i7-6700HQ CPU
Red Hat QXL controller

Afbeelding 6: Why Not Windows 11 AMD Ryzen 7 3700X).

De aangemaakte virtuele machine zit achter een NAT (Network Address Translation) met zijn eigen adresreeks. Omdat internet bereikbaar is, is dit in de meeste gevallen geen probleem. Als je in de adresrange van de host wilt werken, moet je een bridge aanmaken. Dat is een leuke uitdaging voor een volgend artikel.

Q35-chipset

De machine die wij hebben aangemaakt gebruikt de Q35-chipset ([link 6](#)). Die hebben we nodig om Secure Boot te kunnen ondersteunen. Er is ook een belangrijke beperking bij deze chipset: geen ondersteuning voor XP/2000. De standaard chipset bij Gnome-Boxes is i440FX.

Conclusie

Gnome-Boxes maakt het mogelijk om in Linux snel een virtuele machine op te zetten met bijvoorbeeld Windows. Het is dan mogelijk met kopiëren en plakken teksten en afbeeldingen uit te wisselen. Mappen van de hostmachine en USB-apparaten kunnen eenvoudig worden gekoppeld. De aangemaakte virtuele machine benadert de snelheid van de hostmachine.

Bij het installeren van Windows 11 wordt niet getest op de processor. Bij een upgrade van Windows 10 naar 11 is dat wel het geval. Het is nu nog niet duidelijk hoe de overgang tussen de eventuele verschillende versies van Windows 11 gaat plaatsvinden. Het kan dus zijn dat bij een niet-compatibele hostprocessor voor upgrades naar een hogere versie van Windows 11 dezelfde procedure nodig is als voor een upgrade van Windows 10 naar 11. Deze procedure is besproken in het artikel over Gnome-Boxes ([link 4](#)).

Links

1. <https://help.gnome.org/users/gnome-boxes/stable/>
2. <https://www.qemu.org/>
3. https://arstechnica.com/gadgets/2021/06/heres-what-youll-need-to-upgrade-to-windows-11/?utm_brand=arstechnica&utm_source=twitter&utm_social-type=owned&utm_medium=social
4. https://www.compusers.nl/system/files/swb-jaargangen/2022/2022-1/SwB20221_Gnome-Boxes.pdf
5. <https://www.spice-space.org/download.html>