I recently had the opportunity to assist on a project where a partner was using N-Series Azure VMs. My part of this effort was developing a script to automate the setup of the VMs. To perform the VM setup and configuration, an ARM template was used. The ARM template approach was used because doing so provided consistency with several other ARM templates being used for other parts of the project.

Setting up Azure VMs using ARM templates is common. There are many articles, blog posts, and sample templates available to help get started. That isn’t itself especially interesting. The interesting part, at least for me, was the N-Series aspect. N-Series VMs require a separate step to install the NVIDIA driver to take advantage of the GPU capabilities of the VM. There are instructions (<https://docs.microsoft.com/en-us/azure/virtual-machines/windows/n-series-driver-setup>) on how to install the driver, but those instructions assume you like to remote into the VM each time you create a VM, and then run an installation program. That’s tolerable if doing it only a few times. Any more than that, and it’s time for automation.

The v370.12 driver (which is the current version linked via the Azure documentation page) uses a self-extracting file to first extract the setup components to a directory, and then executes the setup program. By scouring a few other blogs on performing a silent install of NVIDIA drivers, I could piece together the necessary switches to provide to the installation program to perform a silent install.

> 370.12\_grid\_win8\_win7\_server2012R2\_server2008R2\_64bit\_international.exe -s -noreboot -clean

This tells the installation program to install silently, to not perform a reboot after the installation is complete, and to perform a clean install (restores all NVIDIA settings to the default values).

Now I need to take that and bake it into a PowerShell script to execute via a custom script extension. By doing so, I can let ARM do its thing by provisioning the VM and related resources (NIC, Virtual Network, IP address, etc.), and then invoke a PowerShell script to install the NVIDIA driver.

The custom script extension will execute a few different steps:

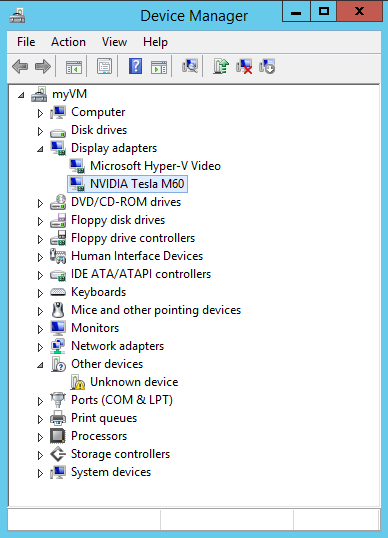
1. Download the NVIDIA driver setup file from Azure Blob storage. I put the setup file in blob storage to make sure that this specific one is the one to be used.
2. Download a PowerShell script which will execute the NVIDIA driver setup program with parameters to do so silently.
3. Wait for the installation program to finish
4. Force a reboot of the VM

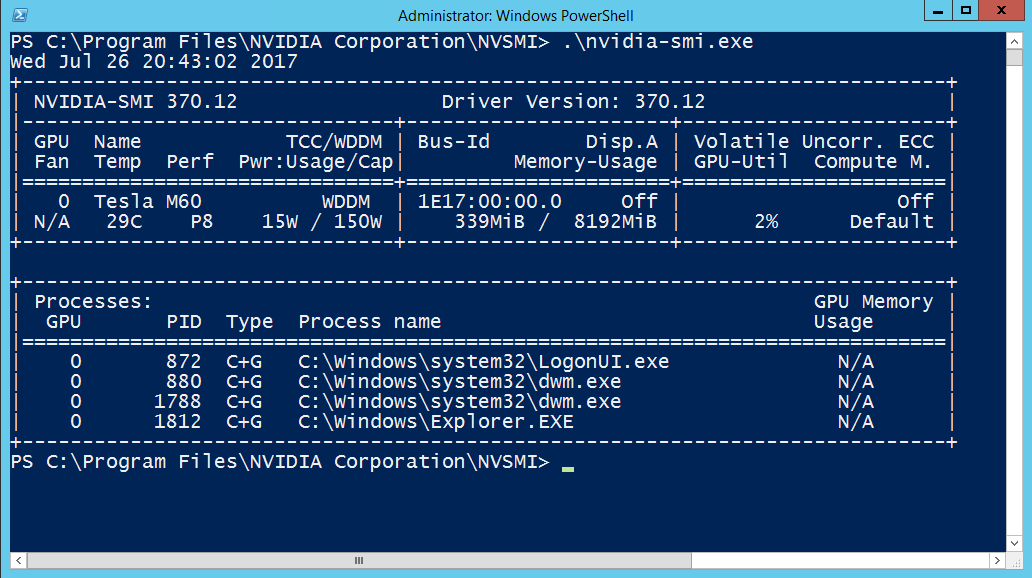
In this scenario, I also need to get the assets used by the custom script extension – the NVIDIA driver setup file and PowerShell script (which will execute the NVIDIA driver setup file) – uploaded to Azure Blob storage. That can easily be accomplished with the same PowerShell script used to deploy the ARM template. That script will perform the following tasks:

1. Create a new resource group
2. Create a new storage account and container
3. Upload the NVIDIA driver setup file and related PowerShell script to the newly created storage account
4. Execute the ARM template

You can find the full ARM template, custom script, and deployment script at <https://github.com/mcollier/setup-nvidia-drivers-azure-vm>.

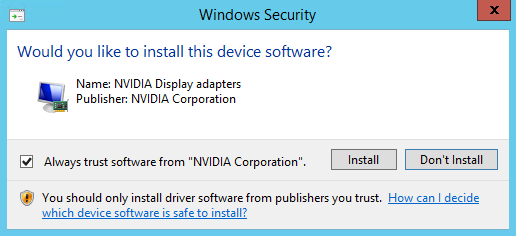
In order to verify it all worked, I can RDP into the VM and verify the driver installation (<https://docs.microsoft.com/en-us/azure/virtual-machines/windows/n-series-driver-setup#verify-driver-installation>).



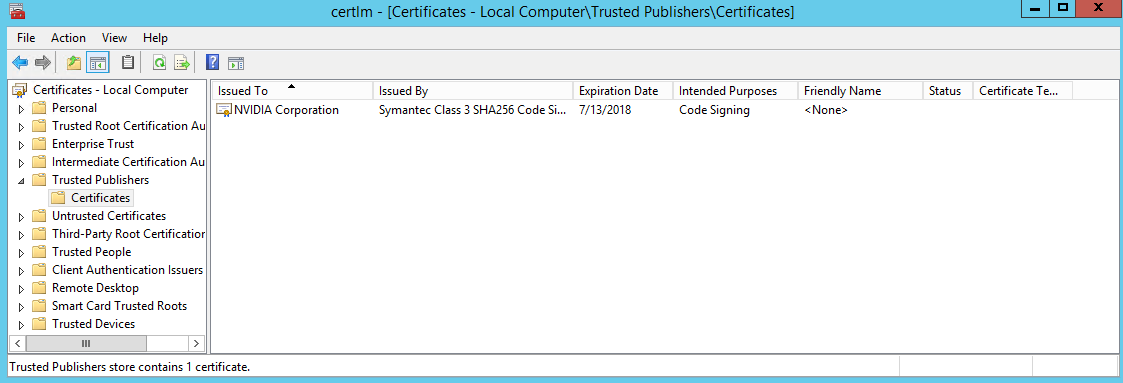


**What about the v369.95 driver?**

The 369.95 version of the NVIDIA driver is slightly different from a setup process. To start with, it is a ZIP file instated of an EXE. Also, it isn’t digitally signed. To use this version of the NVIDIA driver, a few changes to the setup script are necessary. First, the file contents need to be extracted/unzipped. That’s doable via some PowerShell in the script executed via the custom script extension. Getting around the lack of a digitally signed driver is a bit more . . . interesting. If you were to install the driver manually, you would receive a prompt from Windows asking you to confirm that installing the driver is REALLY what is desired.



Completing the manual installation will result in a certificate installed to the VM’s Trusted Publisher certificate store. The certificate can then be exported and saved to Azure Blob storage.



I can use that certificate as part of the automated install process. By using the **certutil.exe** program it is possible to install the certificate into the Trusted Publisher store on a *new* VM. This step can be included in the PowerShell script executed via the custom script extension.

An example of this approach can be found at <https://github.com/mcollier/setup-nvidia-drivers-azure-vm/tree/driver-369.95>.

**Alternative Approach**

An alternative approach is to create a custom VM image with the necessary NVIDIA driver already installed. The advantage with this approach is you don’t have to go through the custom script step. Plus, you can add additional software or configuration as needed. The disadvantage is you’re then accepting responsibility for keeping the VM patched on a regular basis. If you use an image provided by Microsoft, those images are patched on a regular (often at least once per month) basis.

**Resources**:

Custom Script Extension for Windows (<https://docs.microsoft.com/en-us/azure/virtual-machines/windows/extensions-customscript?toc=%2fazure%2fvirtual-machines%2fwindows%2ftoc.json>)

There was previously a comment at <https://docs.microsoft.com/en-us/azure/virtual-machines/windows/n-series-driver-setup> about a silent install for the NVIDIA drivers, which linked to <https://github.com/Azure/azure-accessplatform-windows-gpu/blob/9269ac0665cc6e3ae3cbbece842a8fe2cba12570/CustomScripts/nVIDIAdTeradiciLeostreamAgents.ps1#L75-L92>. That comment appears to be gone.

Silent install of NVIDIA drivers - https://lazyadmin.nl/it/deploy-nvidia-drivers/ and <http://www.itninja.com/question/how-to-install-custom-nvidia-graphic-drivers-silently-in-k1000-managed-installations>.