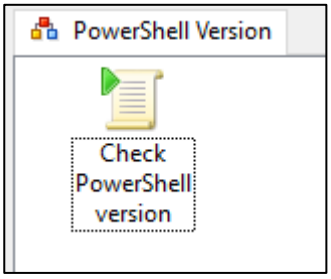
Use native PowerShell version in Orchestrator

By Leon Laude - leon.laude@gmail.com

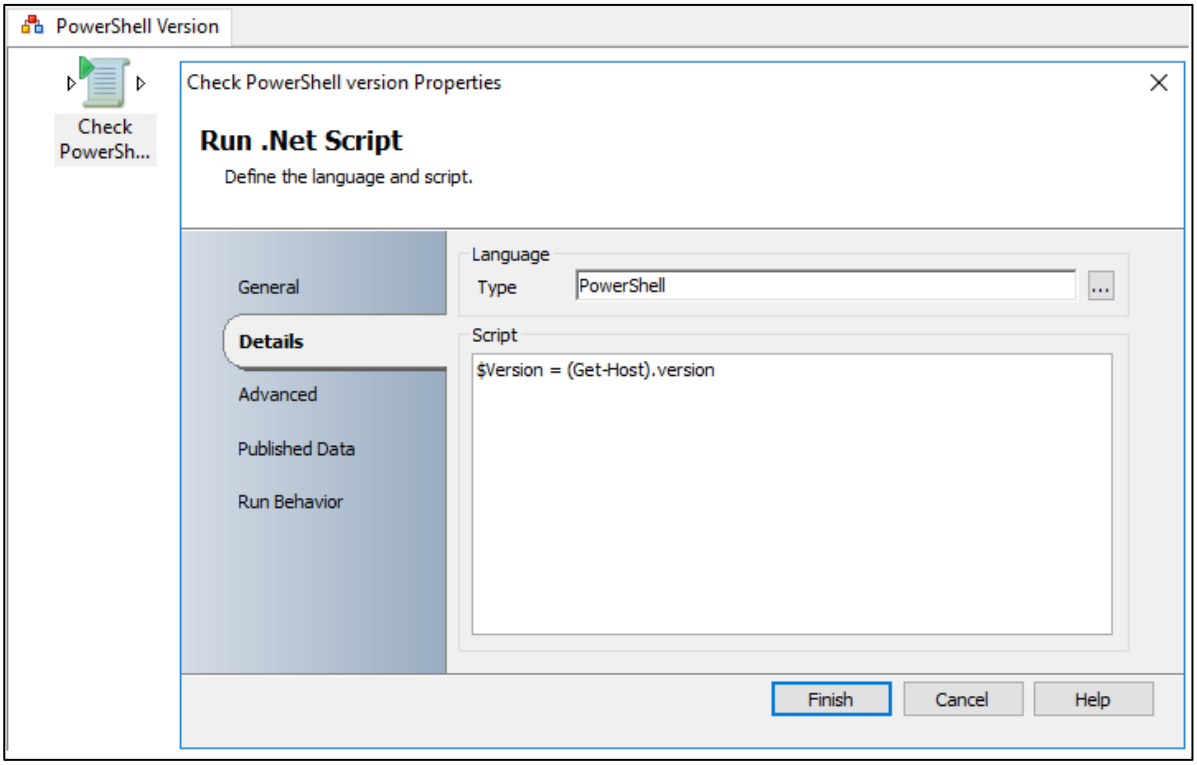
2018-05-01

Let's make sure first what version of PowerShell Orchestrator is using before making any changes.

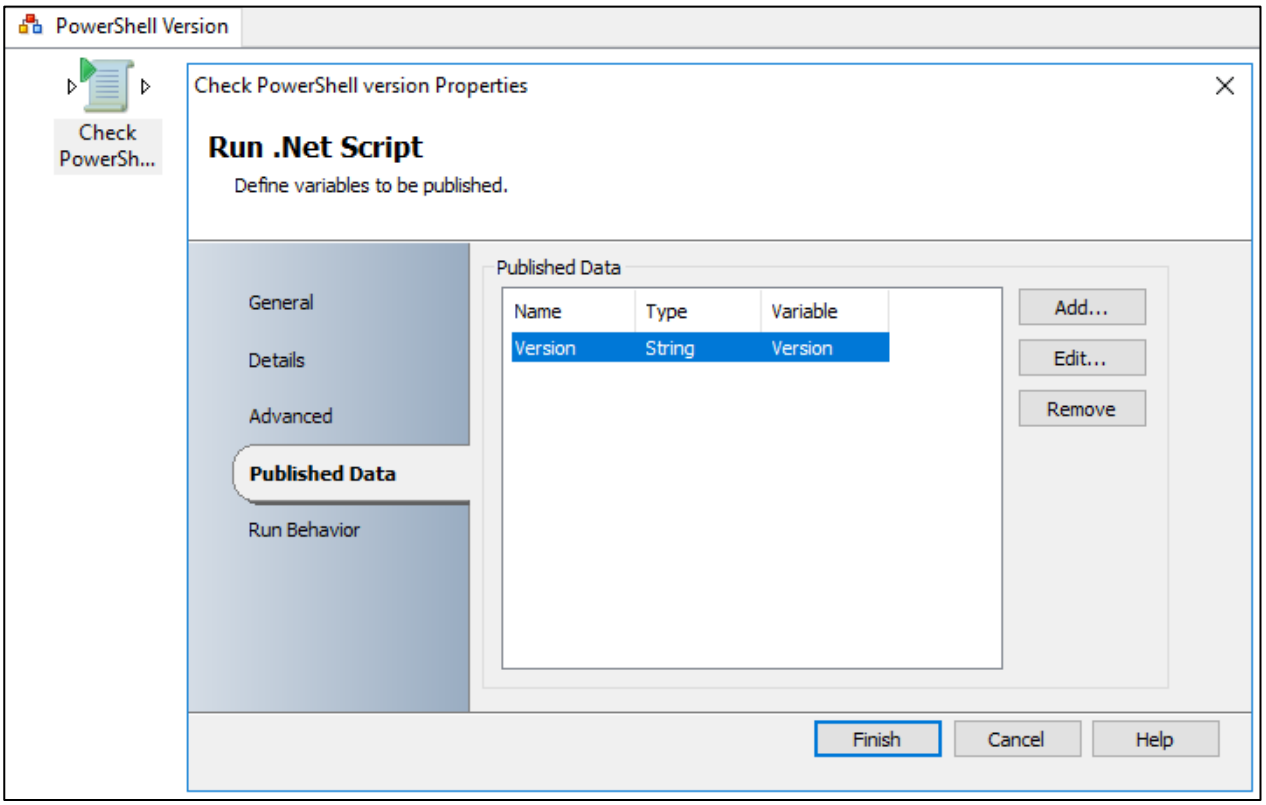
1. Create a runbook.
2. Add a **Run .Net Script** activity which can be found under the System activities.



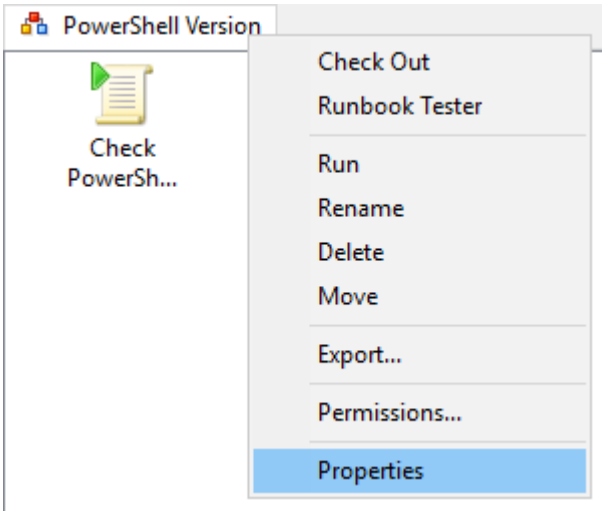
3. Edit the Run .Net Script activity and change the Type to **PowerShell**
4. In the Script block we want to make a variable that will contain the PowerShell version.



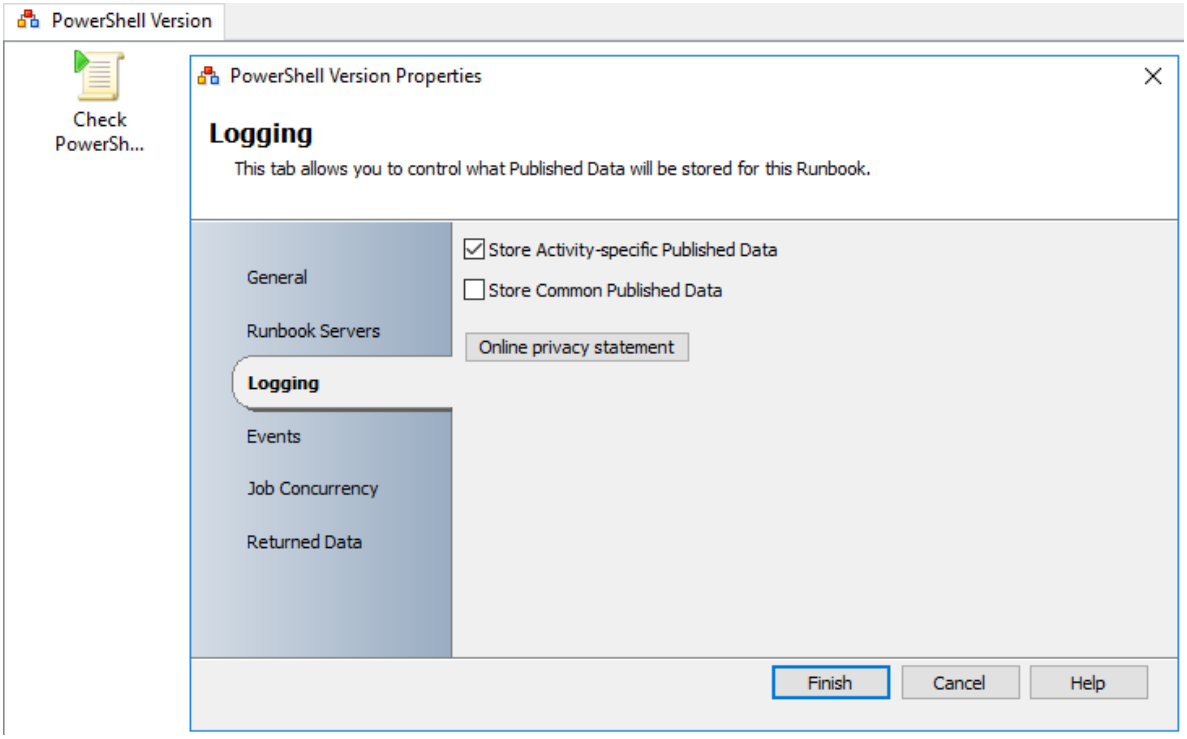
5. Navigate to **Published Data** to create the same named variable (**String** variable) as you did in the script block and then press **Finish**.



6. To get the information about this variable we will need to enable extended logging, you can do this by **right clicking** your runbook and then choose **Properties**.

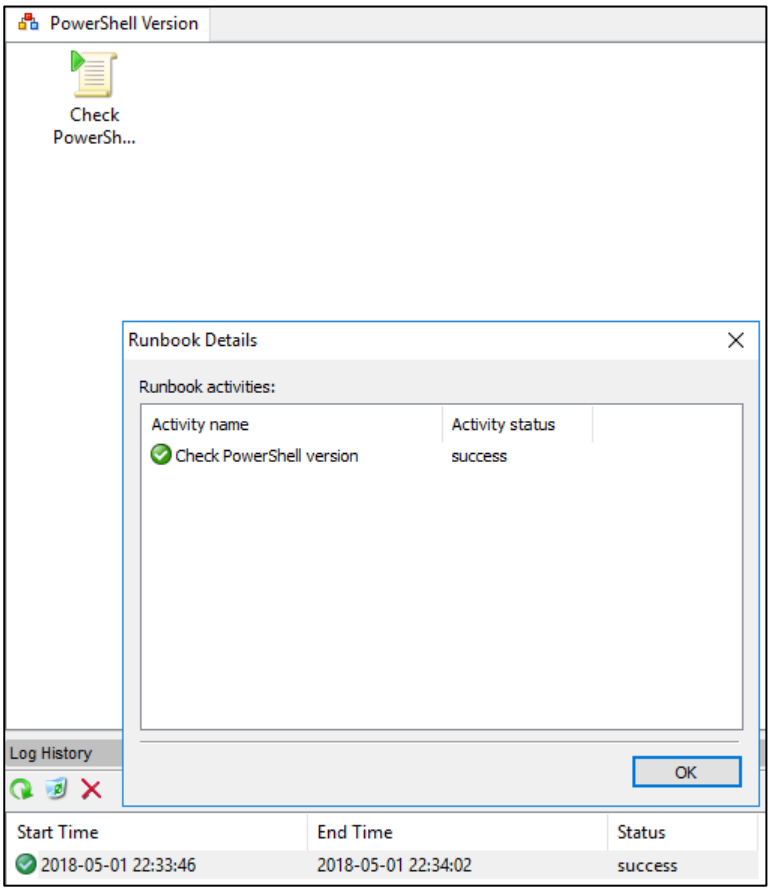


7. Then navigate to the **Logging** tab, make sure to check the following box: **Store Activity-specific Published Data** and then press **Finish**.

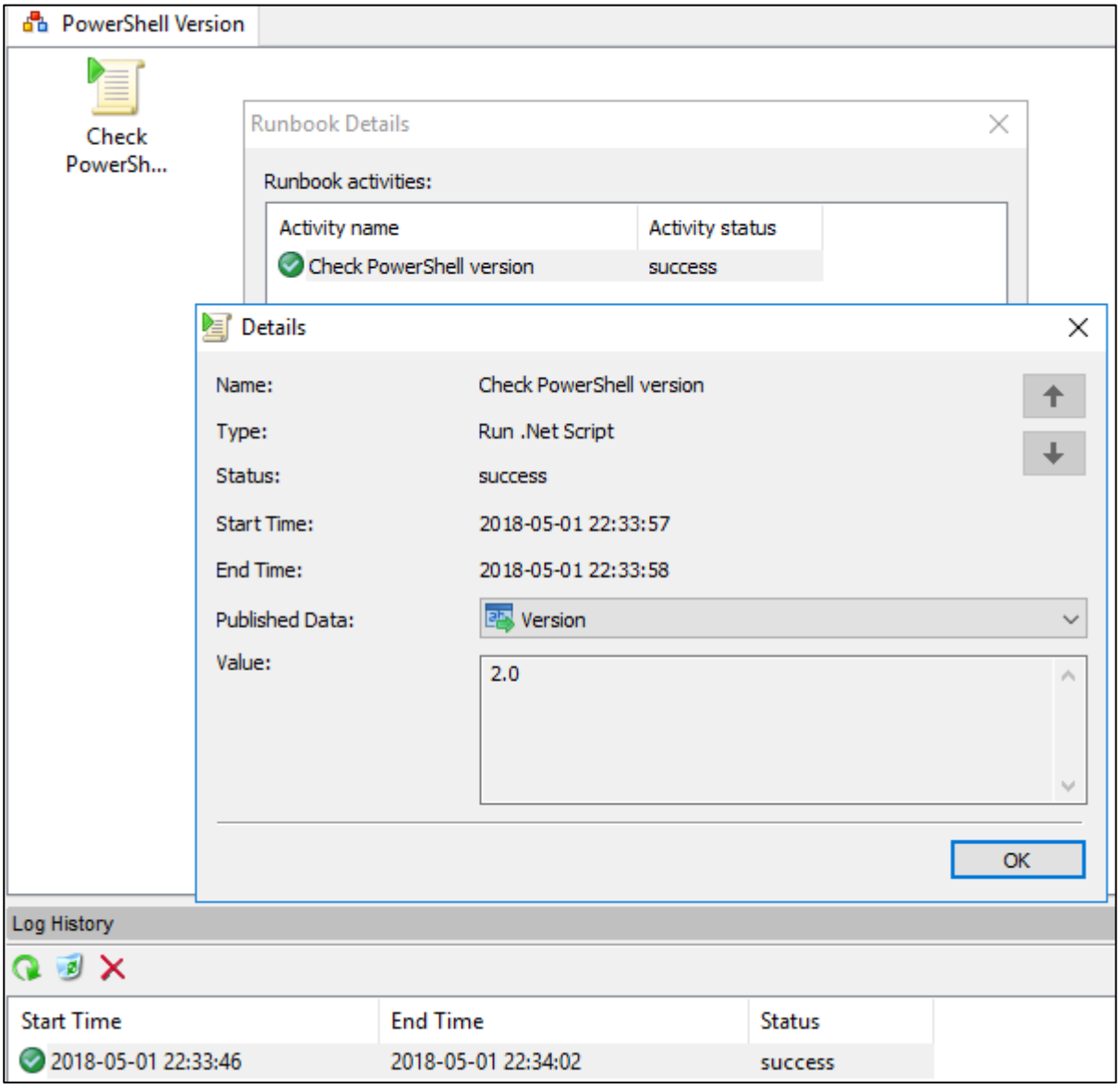


8. Run your runbook once, when the runbook has run you should see Successful event in the **Log History** at the bottom.

9. Now open the log by **double clicking** the log entry in the **Log History**, a new window should open up.



10. **Double click** the entry again and scroll down to the bottom until you find your variable name.

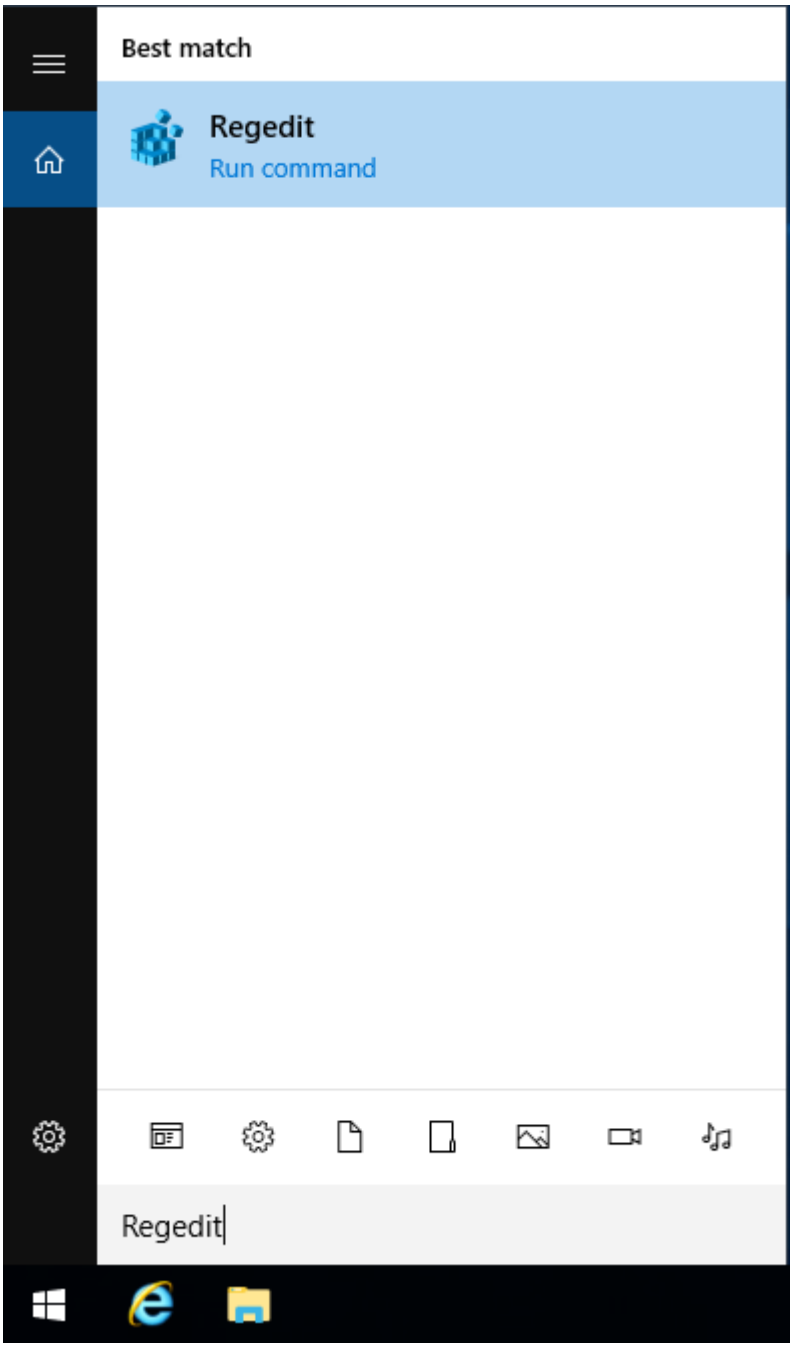


11. As you can see the value is displayed as **2.0** which means Orchestrator is currently running PowerShell version 2.0.

Now let's go through the first method to force Orchestrator to use Window's native PowerShell version.

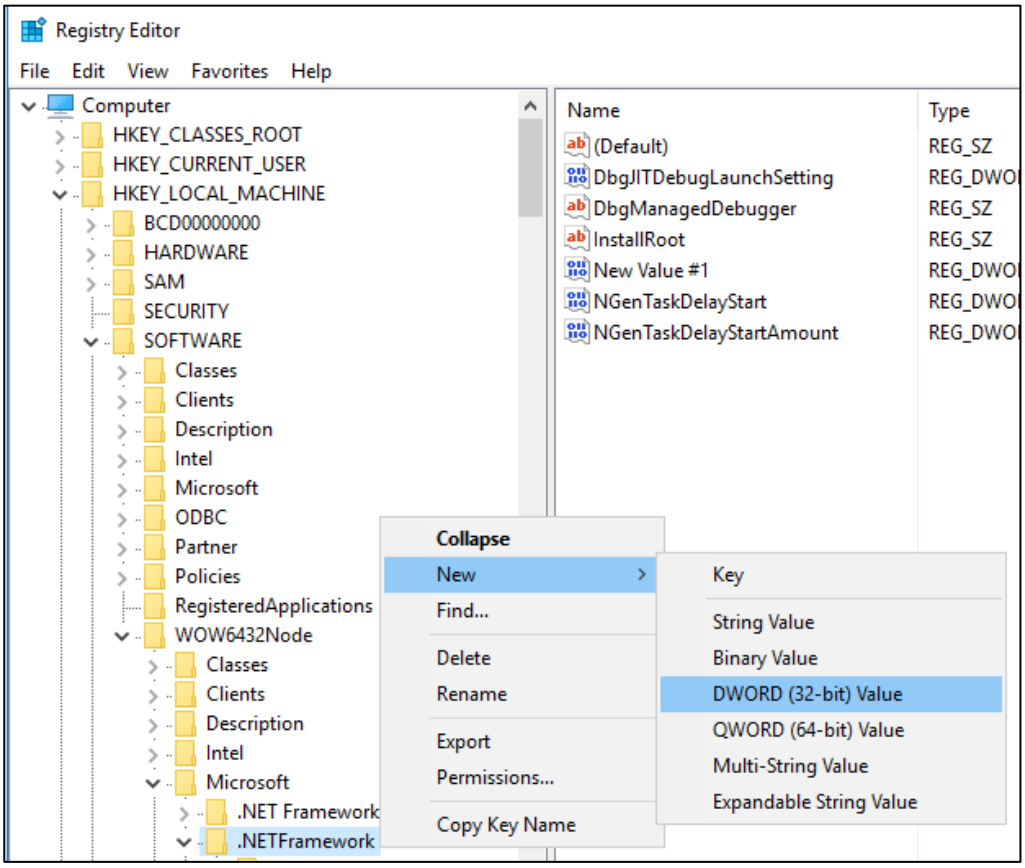
Method 1

1. Go to **Start** and then type **Regedit** and press **Enter**.

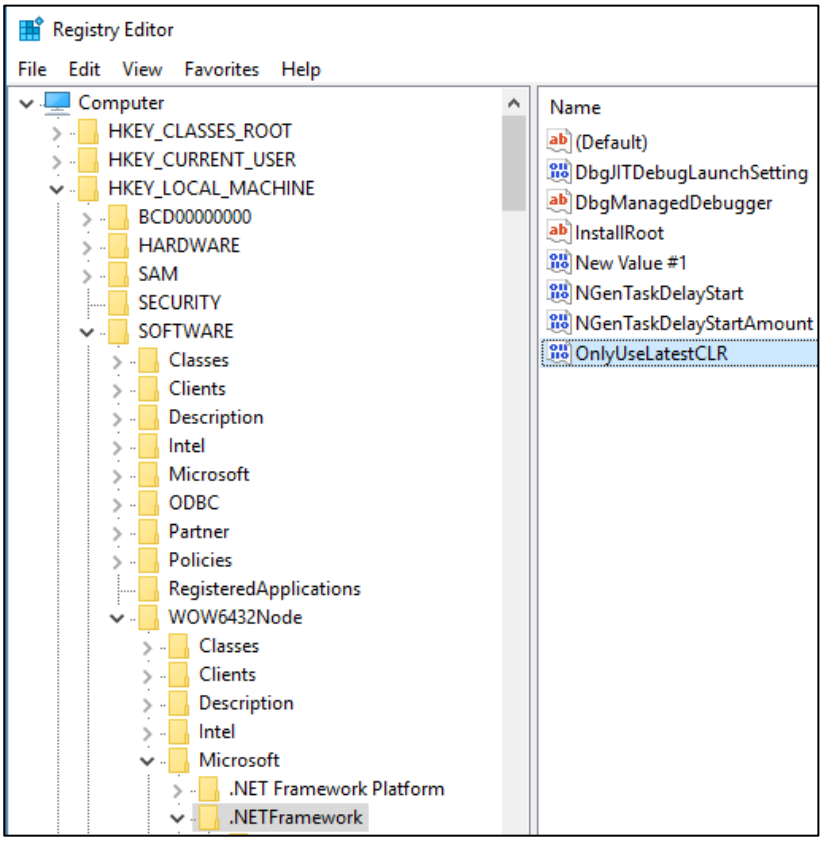


2. Go to the following registry path: **HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Microsoft\.NETFramework**

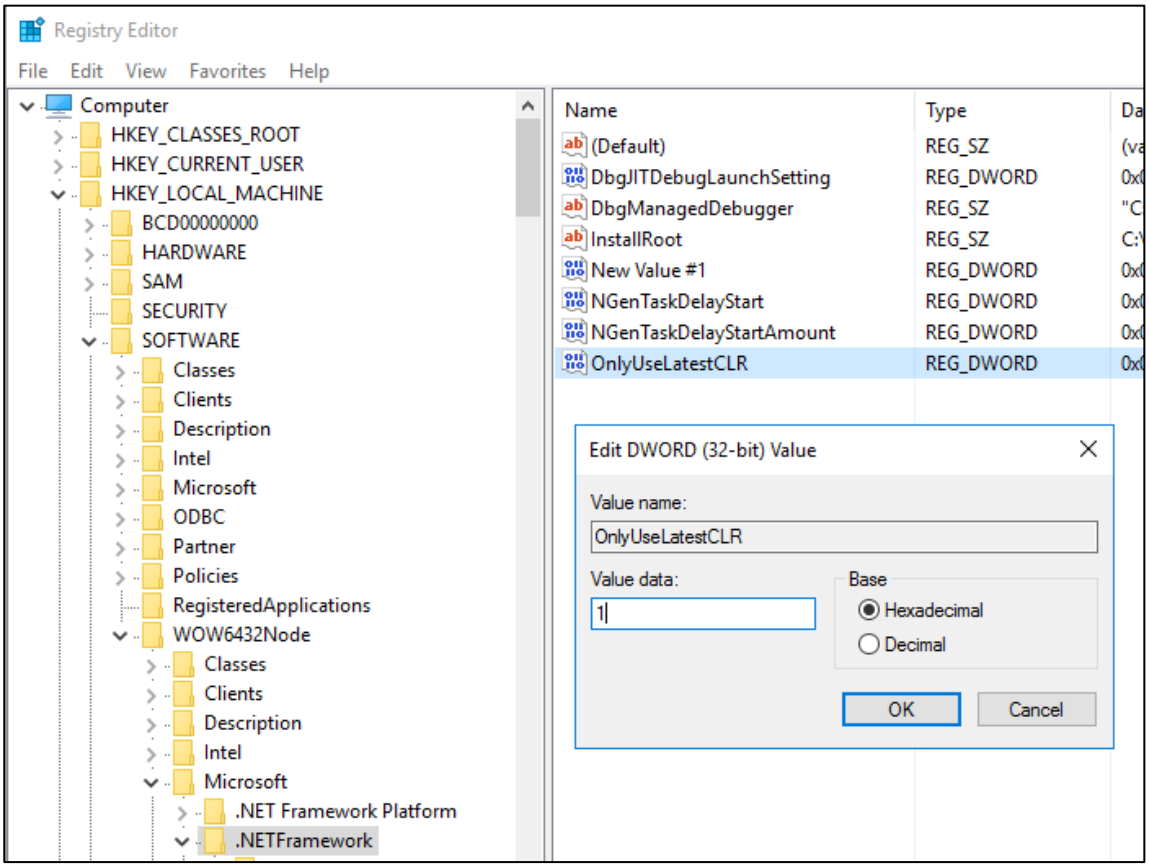
3. Create a new entry by right clicking the .NETFramework folder and choose **New** -> **DWORD (32-bit) Value**.



4. Set the value name to: **OnlyUseLatestCLR**



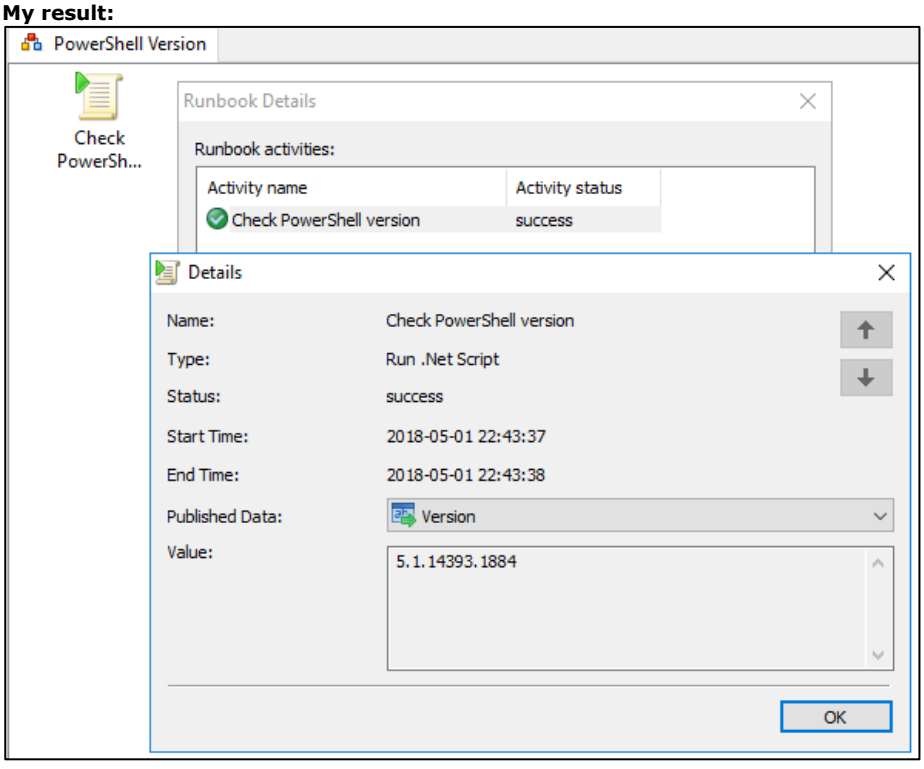
5. Set the value data to: **1**



6. Now restart your Orchestrator server.

7. Once you have restarted your Orchestrator server, re run your runbook that you previously created and see what version it will give

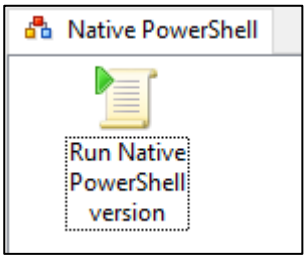
My result:



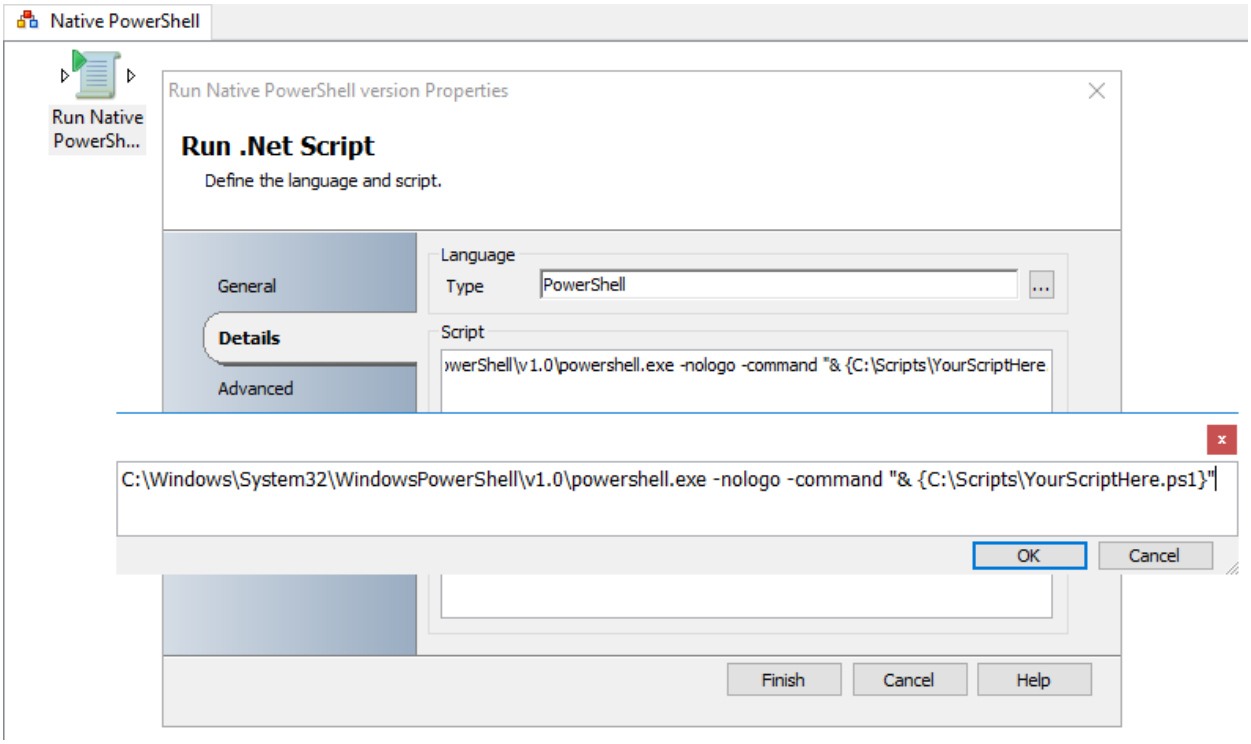
The second way of making Orchestrator use the Windows native PowerShell version is by calling PowerShell from within the **Run .Net Script** activity.

Method 2

1. Create a runbook.
2. Add a **Run .Net Script** activity which can be found under the System activities.



3. Edit the Run .Net Script activity and change the Type to **PowerShell**
4. In the Script block we will call powershell.exe from outside by writing the following: **C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe -nologo -command "&{C:\Scripts\YourScriptHere.ps1}"**



5. Change the path to correspond where your PowerShell scripts are located and then insert your script name at the end.
6. Now just **Check In** your runbook and run it!