Relativity Dev VM

Running Vagrant and Chef Scripts

Documentation

[October 16, 2019]

Table of Contents

[1 Requirements 3](#_Toc510011062)

[2 Downloads 3](#_Toc510011063)

[2.1.1 Relativity installation 4](#_Toc510011064)

[2.1.2 SQL Server 2016 Developer Edition 4](#_Toc510011065)

[2.1.3 Service Bus 1.1 Defect Windows Update 4](#_Toc510011066)

[2.1.4 Install Visual Studio Code text editor 5](#_Toc510011067)

[3 Setup Workstation 5](#_Toc510011068)

[3.1 Login 5](#_Toc510011069)

[3.2 Setup PowerShell 5](#_Toc510011070)

[3.3 Install Chocolatey 7](#_Toc510011071)

[3.4 Install GIT Version 2.14.1 7](#_Toc510011072)

[3.5 Clone Dev VM repository from Github 8](#_Toc510011073)

[3.6 Install Chef Development Kit Version 2.4.17 8](#_Toc510011074)

[3.7 Enable the Hyper-V Windows feature 9](#_Toc510011075)

[3.8 Setup Hyper-V Virtual Switch 12](#_Toc510011076)

[3.9 Share Internet Connection 15](#_Toc510011077)

[3.10 Install Vagrant Version 2.0.1 17](#_Toc510011078)

[3.11 Install Vagrant Reboot plugin 17](#_Toc510011079)

[3.12 Install Vagrant berkshelf plugin 18](#_Toc510011080)

[3.13 Add Dev VM Windows Base machine to Vagrant boxes list 18](#_Toc510011081)

[4 Create New Dev VM 19](#_Toc510011082)

[4.1 Login 19](#_Toc510011083)

[4.2 Navigate to the Dev VM GIT repository 19](#_Toc510011084)

[4.3 Switch to vagrant-poc branch 19](#_Toc510011085)

[4.4 Copy the config.yaml file required for Dev VM 19](#_Toc510011086)

[4.5 Update Configuration file 22](#_Toc510011087)

[4.6 Run the Dev VM Creation script 22](#_Toc510011088)

[4.6.1 Navigate to the Dev VM GIT repository 22](#_Toc510011089)

[4.6.2 List all the files in Chef folder 23](#_Toc510011090)

[4.6.3 Run the Creation script 23](#_Toc510011091)

[5 Clean Workstation 24](#_Toc510011092)

[5.1 Login 24](#_Toc510011093)

[5.2 Remove Vagrant Box Image 24](#_Toc510011094)

[5.3 Uninstall Vagrant berkshelf plugin 24](#_Toc510011095)

[5.4 Uninstall Vagrant reload plugin 24](#_Toc510011096)

[5.5 Uninstall Vagrant 24](#_Toc510011097)

[5.6 Uninstall Chocolatey 24](#_Toc510011098)

[5.7 Remove Hyper-V Virtual Switch 25](#_Toc510011099)

[5.8 Uninstall Hyper-V 25](#_Toc510011100)

[5.9 Uninstall Chef Development Kit 25](#_Toc510011101)

# Requirements

* CPU cores: 4 cores
* RAM: 12GB
* Storage: 140GB
* Licenses
  + Windows Server 2012 R2
  + Relativity
* Windows 10 Pro with Hyper-V features enabled.
* Windows Admin account.
* Internet connection

# Downloads

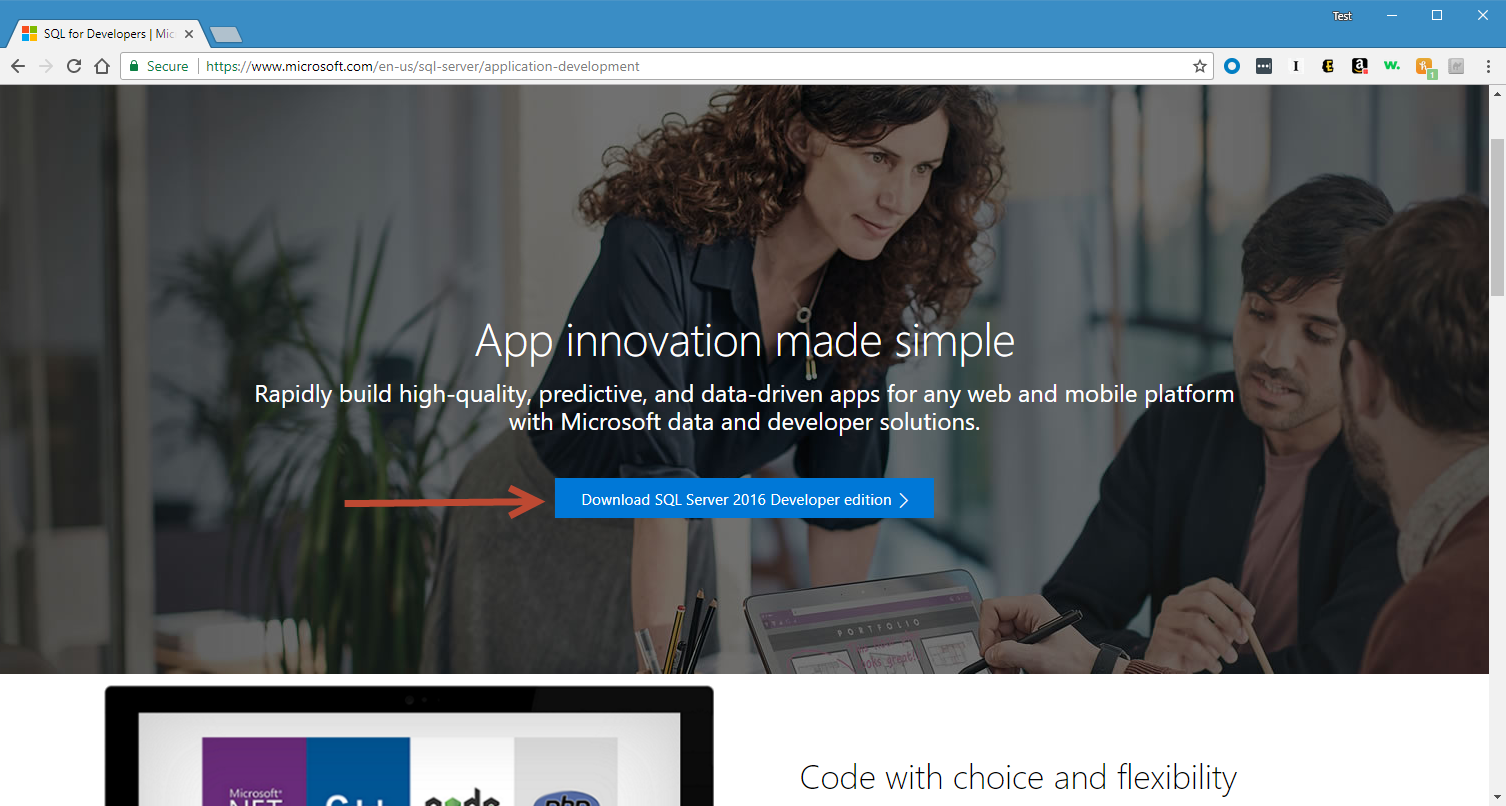
* List of files needed for running the Vagrant/Chef recipes.
  + Relativity installer
  + Invariant installer
  + Secret Store
  + Datagrid
  + SQL Server 2016 Developer Edition
  + Service Bus Defect Windows update
  + Visual Studio Code text editor
  + MS Office Professional Plus 2010
  + MS Works Converter
  + Lotus Notes
  + Jungum

### Relativity installation

* Please contact [support@relativity.com](mailto:support@relativity.com) to get the installer files for the following:
  + Relativity
  + Invariant
  + Secret Store
  + Datagrid

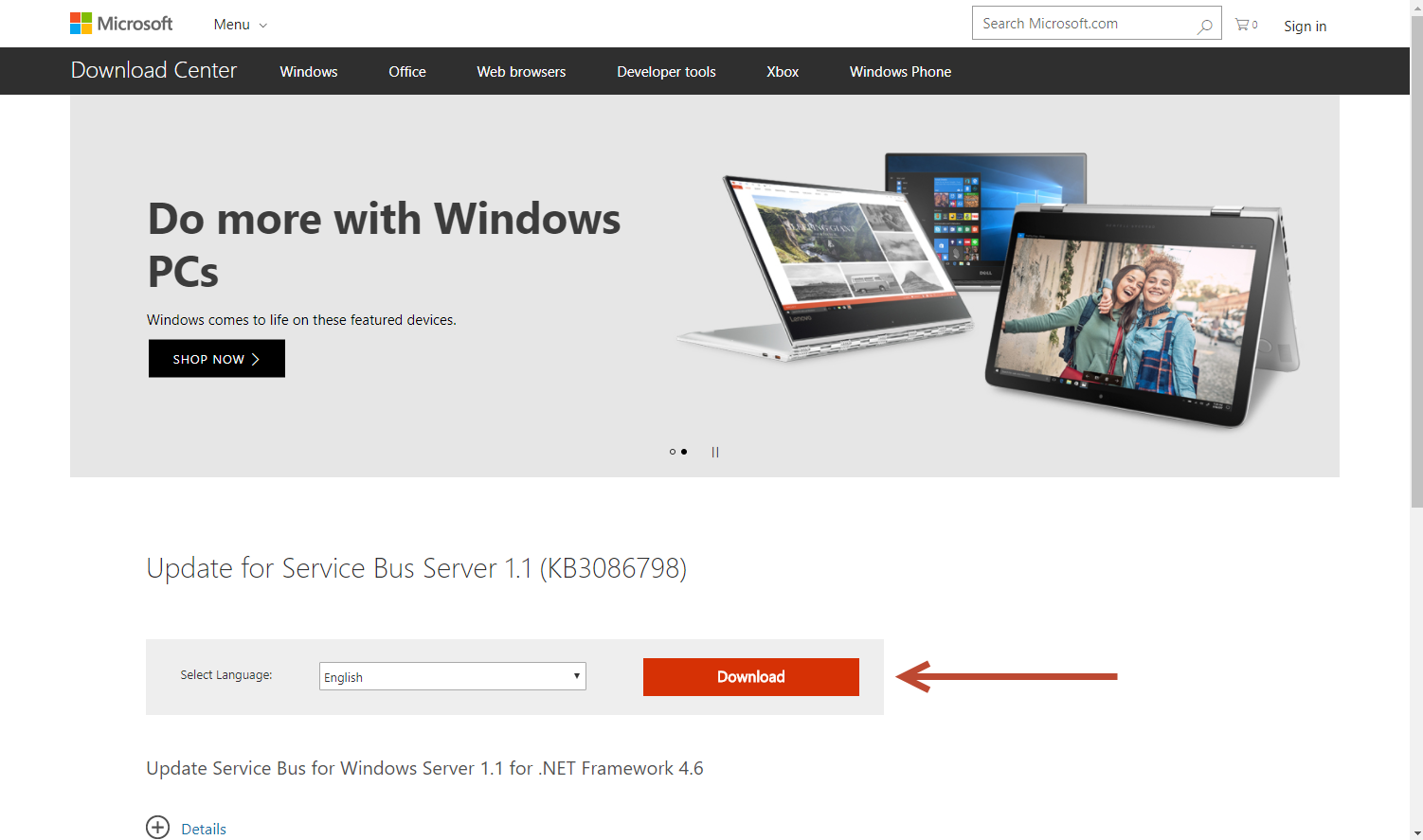
### SQL Server 2016 Developer Edition

* You can download it at this link - <https://www.microsoft.com/en-us/sql-server/application-development>



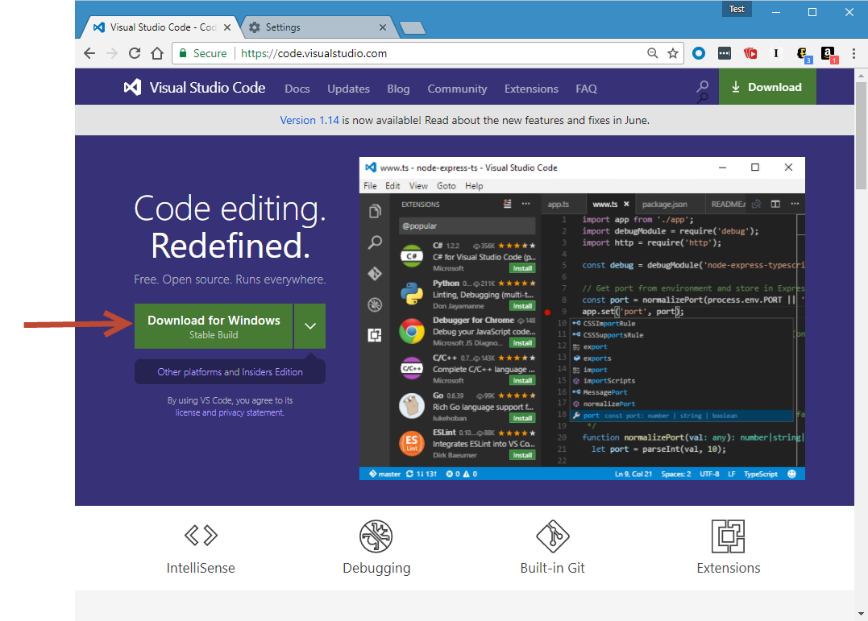
### Service Bus 1.1 Defect Windows Update

* You can download it at this link - <https://www.microsoft.com/en-us/download/details.aspx?id=49496>



### Install Visual Studio Code text editor

* Download and Install Visual Studio Code from this link [Visual Studio Code](https://code.visualstudio.com/)



# Setup Workstation

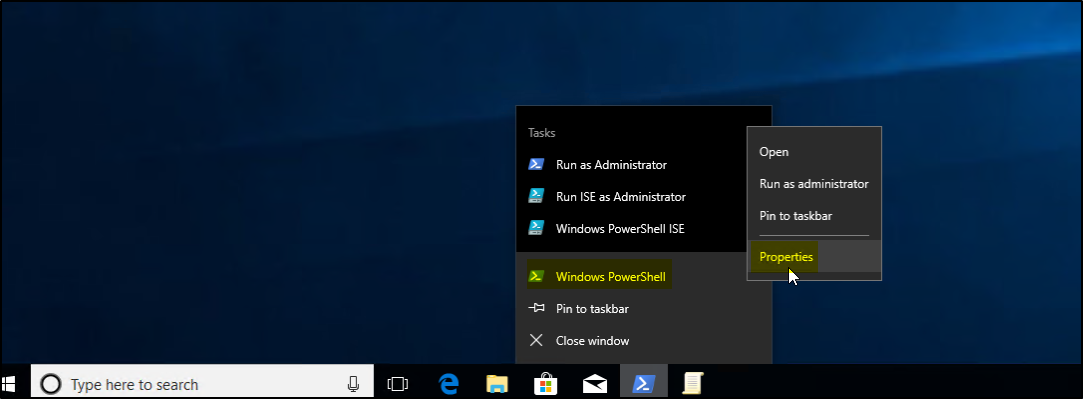
* It’s important that you run the below steps in order.

## Login

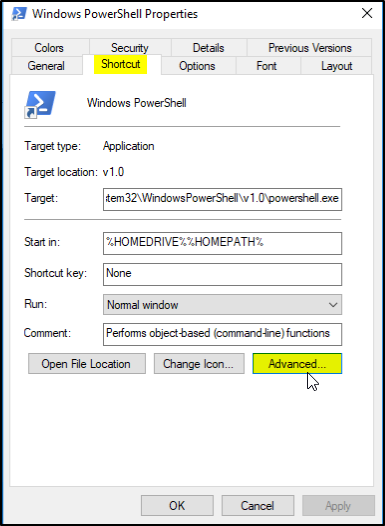
* Login into your workstation with a Windows Administrator account.

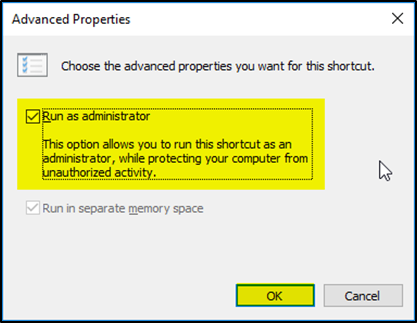
## Setup PowerShell

* Let’s configure PowerShell to always open as an Administrator.
* Pin **PowerShell** to your Task Bar
* Right click on **Windows PowerShell** then **Properties**



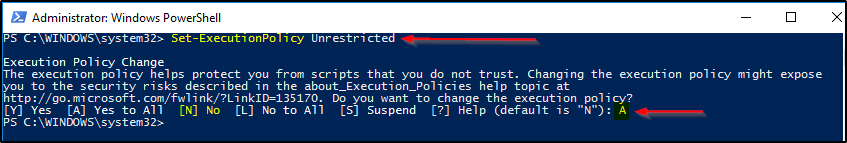
* Find the tab called **Shortcut**, then **Advanced**, then **Run As Administrator**





* Open a PowerShell window (should be opening as an Administrator now) and run the following command:

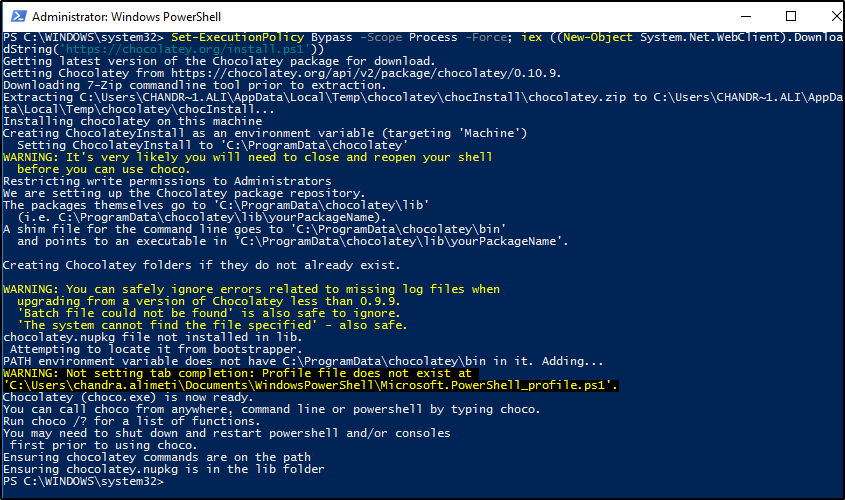
Set-ExecutionPolicy Unrestricted



## Install Chocolatey

* Run the following command in PowerShell window.

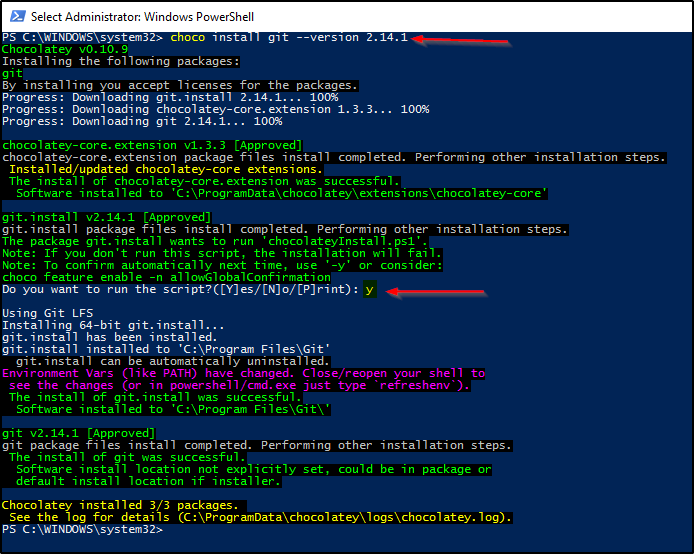
Set-ExecutionPolicy Bypass -Scope Process -Force; iex ((New-Object System.Net.WebClient).DownloadString('https://chocolatey.org/install.ps1'))



## Install GIT Version 2.14.1

* Run the following command in PowerShell window.

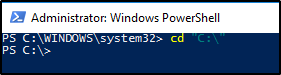
choco install git --version 2.14.1



## Clone Dev VM repository from Github

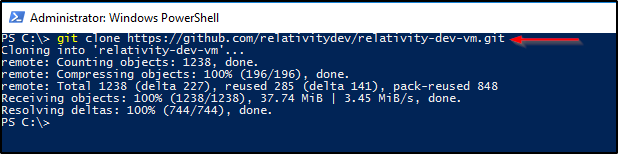
* Run the follo
* wing command in PowerShell window.

cd "C:\"



* Run the following command in PowerShell window.

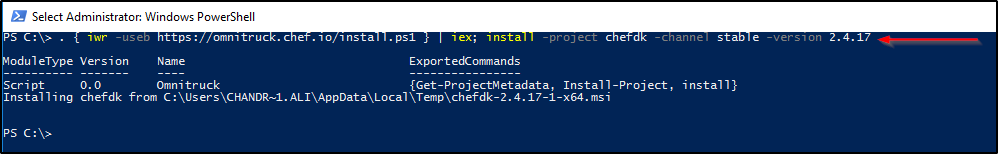
git clone https://github.com/relativitydev/relativity-dev-vm.git



## Install Chef Development Kit Version 2.4.17

* Run the following command in PowerShell window.

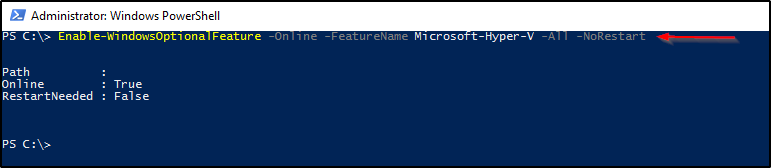
. { iwr -useb https://omnitruck.chef.io/install.ps1 } | iex; install -project chefdk -channel stable -version 2.4.17



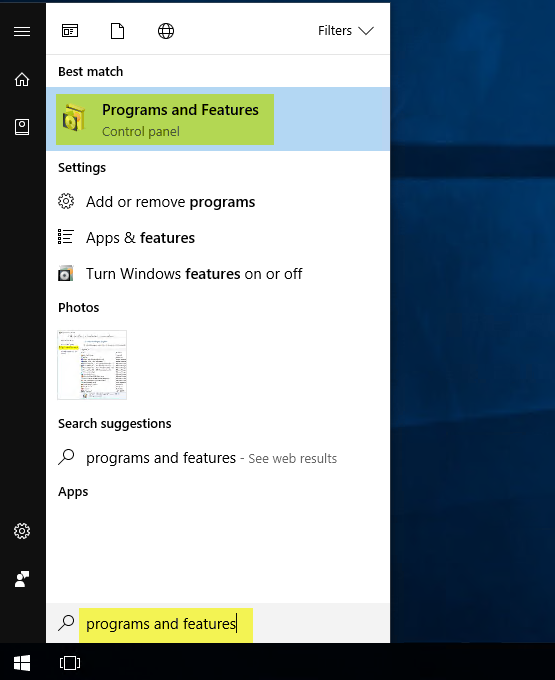
## Enable the Hyper-V Windows feature

* Run the following command in PowerShell window.

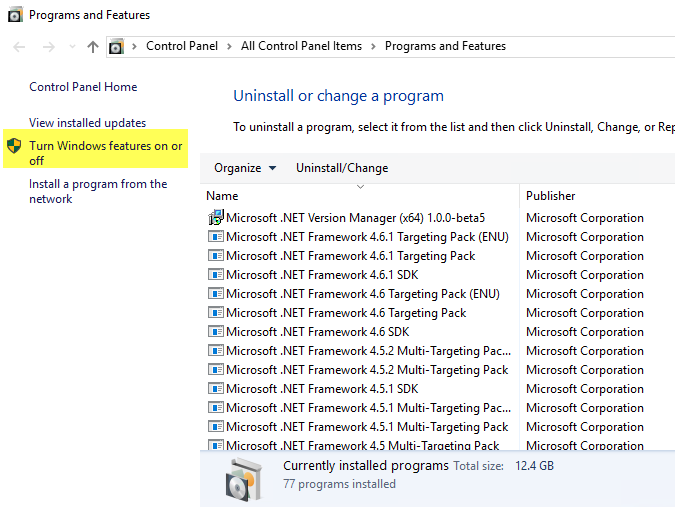
Enable-WindowsOptionalFeature -Online -FeatureName Microsoft-Hyper-V -All -NoRestart



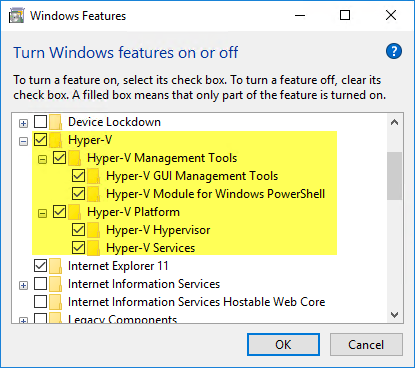
* **Restart your workstation.**
* Verify Windows Hyper-V feature installation
  + Once your workstation finished restart, Go to **Programs and Features**



* + Click on **Turn Windows features on or off**

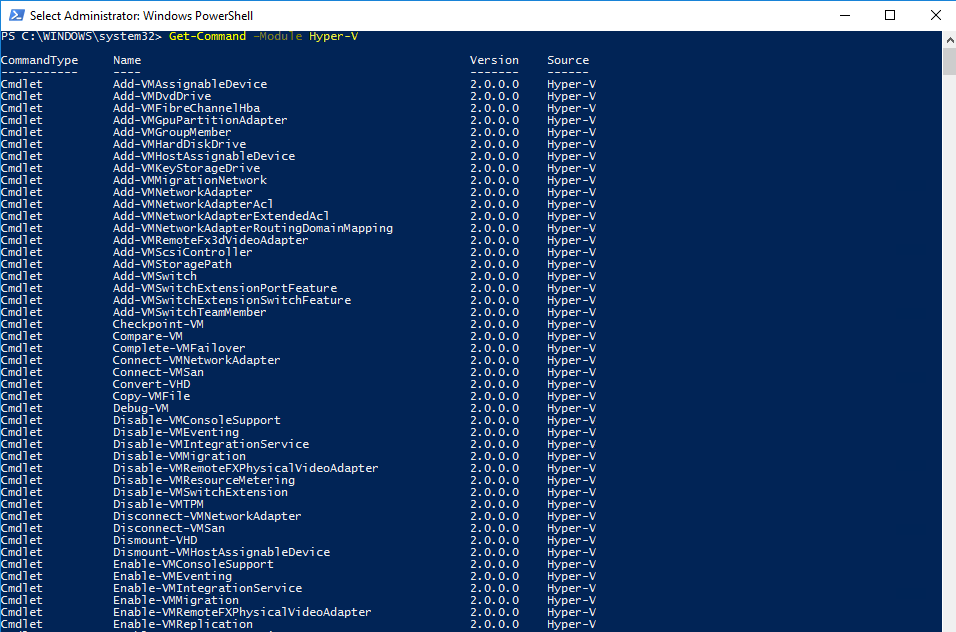


* + Verify if Hyper-V is checked as shown in the below screenshot



* + You can also verify Hyper-V installation via PowerShell
    - Run the following command in PowerShell window. You should see results as shown in the below screenshot.

Get-Command –Module Hyper-V



## Setup Hyper-V Virtual Switch

* Open the **Hyper-V Manager** application.



* Make sure you have an **External Virtual Switch** which lets you access your VM from your local machine.
  + Select your local machine from the left pane.
  + In the Actions pane located in **Hyper-V Manager**, click on the **Virtual Switch Manager** link, see screenshot below.



* + If you don’t already have an external network switch please click the **New virtual network switch** link under **Virtual Switches**. Select **External** for the type of switch and click on the **Create Virtual Switch** button.



* + Provide a friendly name for the new switch (For example: **RelativityDevVmSwitch**) and click **Apply** button.



* + If you get a warning message (see screenshot below), please click the **Yes** button.

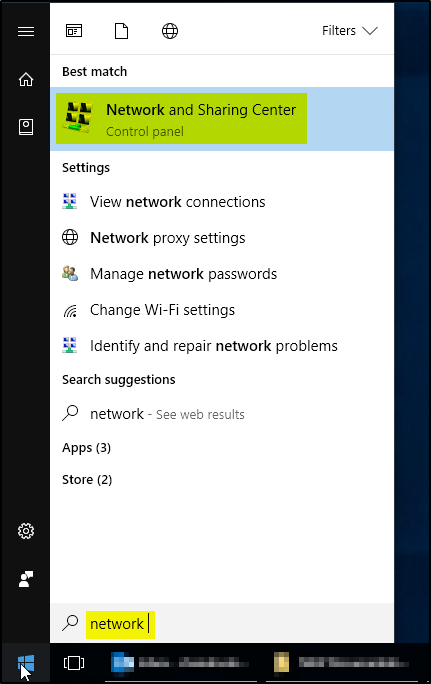


* + Next click the **OK** button to create your new virtual switch.

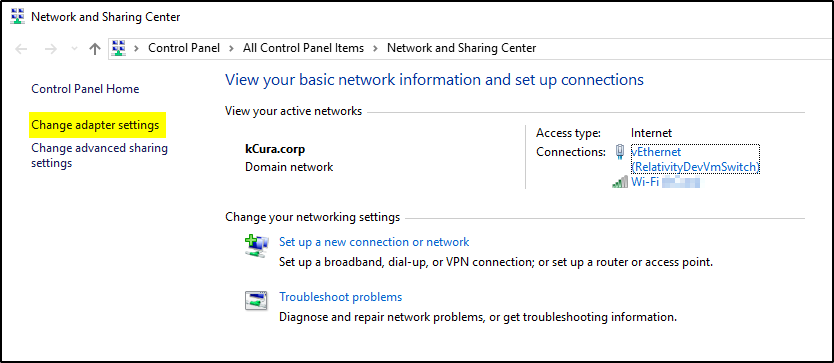


## Share Internet Connection

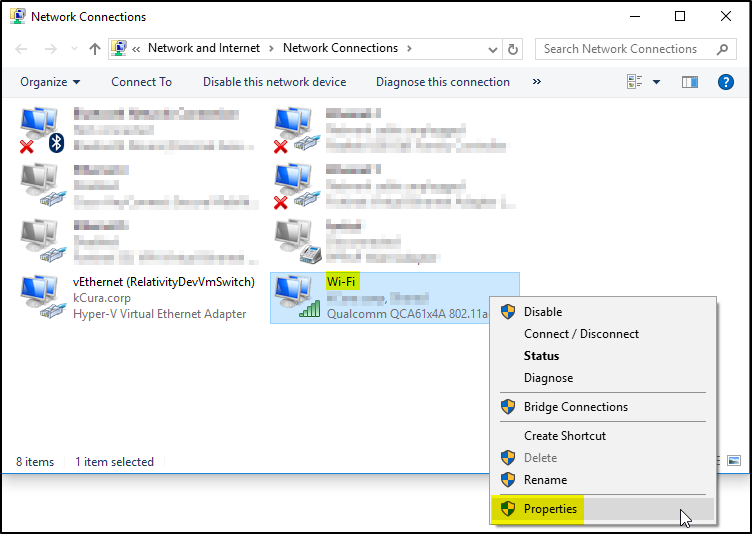
* Sharing the internet connection of your workstation with the Dev VM will make it easy to access it.
* Open **Network and Sharing Center** application on your workstation.



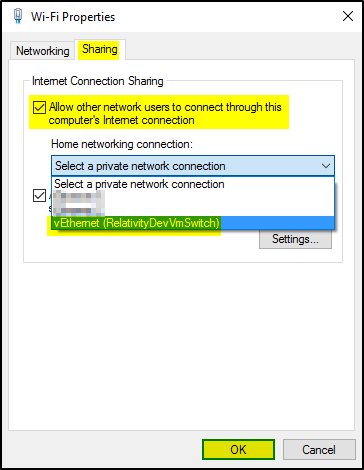
* Click on **Change adapter settings** link on the left pane.



* Right click on the Network connection which has the internet access and select **Properties**.



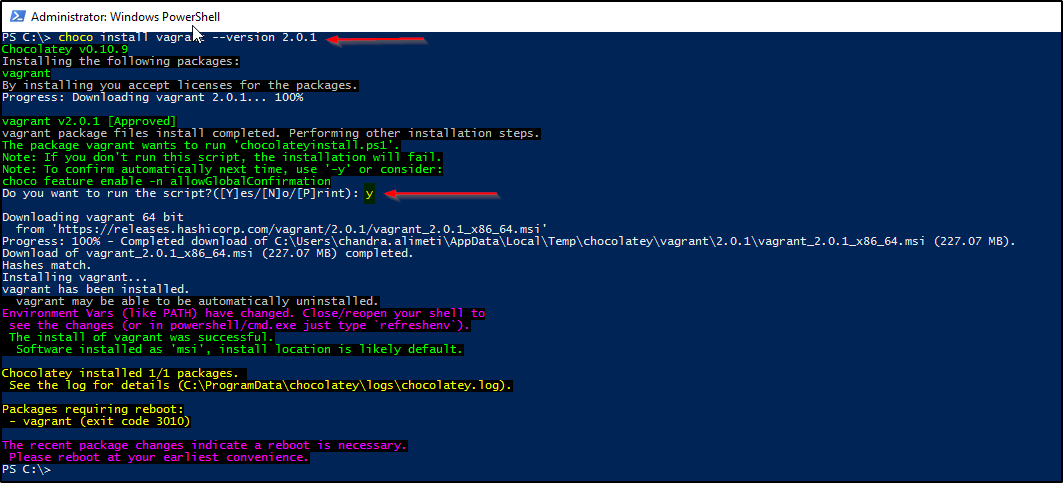
* Select **Sharing** tab, select the checkbox for **Allow other network users to connect through this computer’s Internet connection** option. From the dropdown, select the virtual switch you created for Dev VM and click OK.



## Install Vagrant Version 2.0.1

* Run the following command in PowerShell window.

choco install vagrant --version 2.0.1

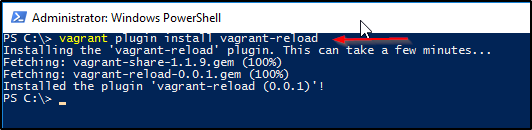
****

* **Restart your workstation.**

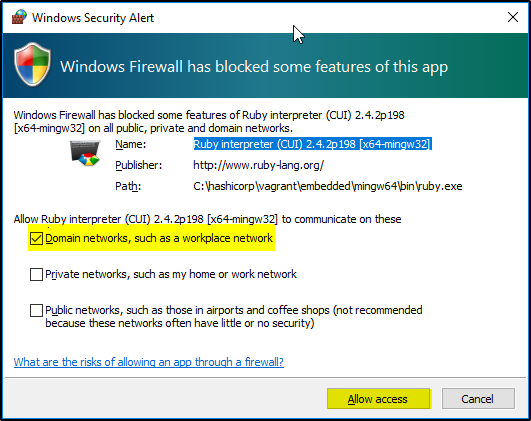
## Install Vagrant Reboot plugin

* Run the following command in PowerShell window.

vagrant plugin install vagrant-reload



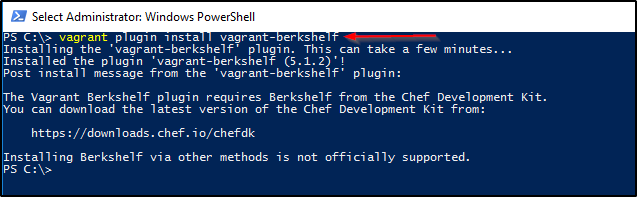
* If you see a popup, select **Allow access** as shown in below screenshot.t



## Install Vagrant berkshelf plugin

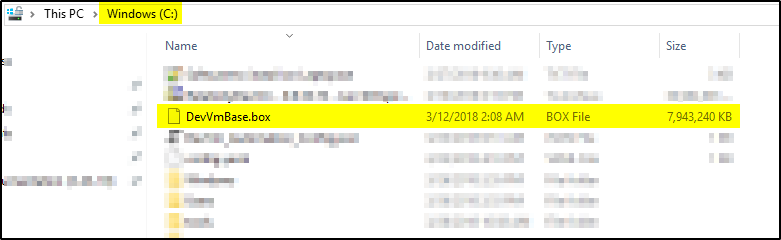
* Run the following command in PowerShell window.

vagrant plugin install vagrant-berkshelf



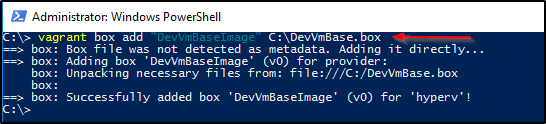
## Add Dev VM Windows Base machine to Vagrant boxes list

* Follow the instructions provided in a different document to create a Windows Base machine. This Windows Base image is needed to run the Vagrant/Chef scripts.
* Once you have the Windows Base box image ready, copy it to your C drive.



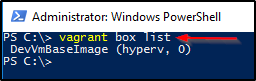
* Next we need to add the Windows Base box image to the Vagrant boxes list. Run the following command in PowerShell window.

vagrant box add "DevVmBaseImage" [C:\DevVmBase.box](file:///\\kcura.corp\shares\Development\DevEx\DevVm_Install_Files\DevVmBase.box)



* Confirm the vagrant box was successfully added. Run the following command in PowerShell window.

vagrant box list



# Create New Dev VM

* It’s important that you run the below steps in order.

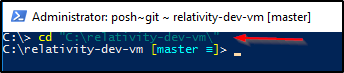
## Login

* Login into your workstation with a Windows Administrator account.

## Navigate to the Dev VM GIT repository

* Run the following command in PowerShell window.

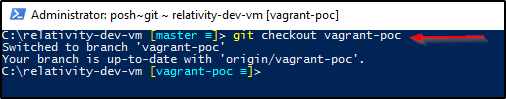
cd "C:\relativity-dev-vm\"



## Switch to vagrant-poc branch

* Run the following command in PowerShell window.

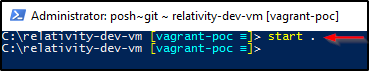
git checkout vagrant-poc



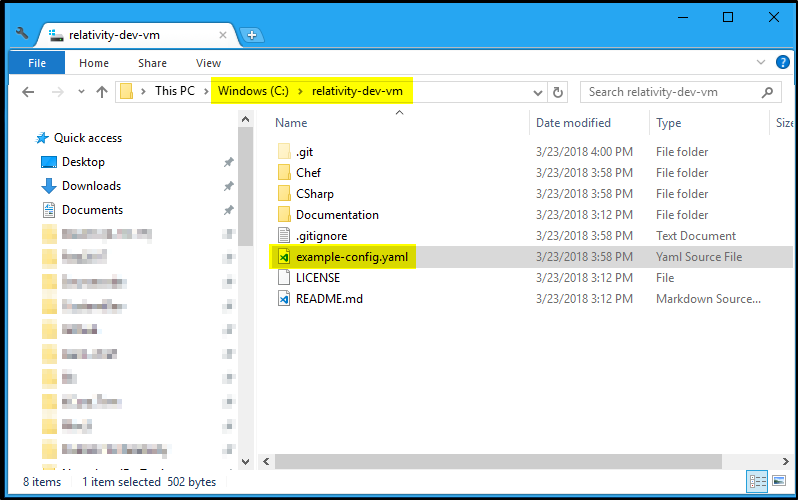
## Copy the config.yaml file required for Dev VM

* Run the following command in PowerShell window.

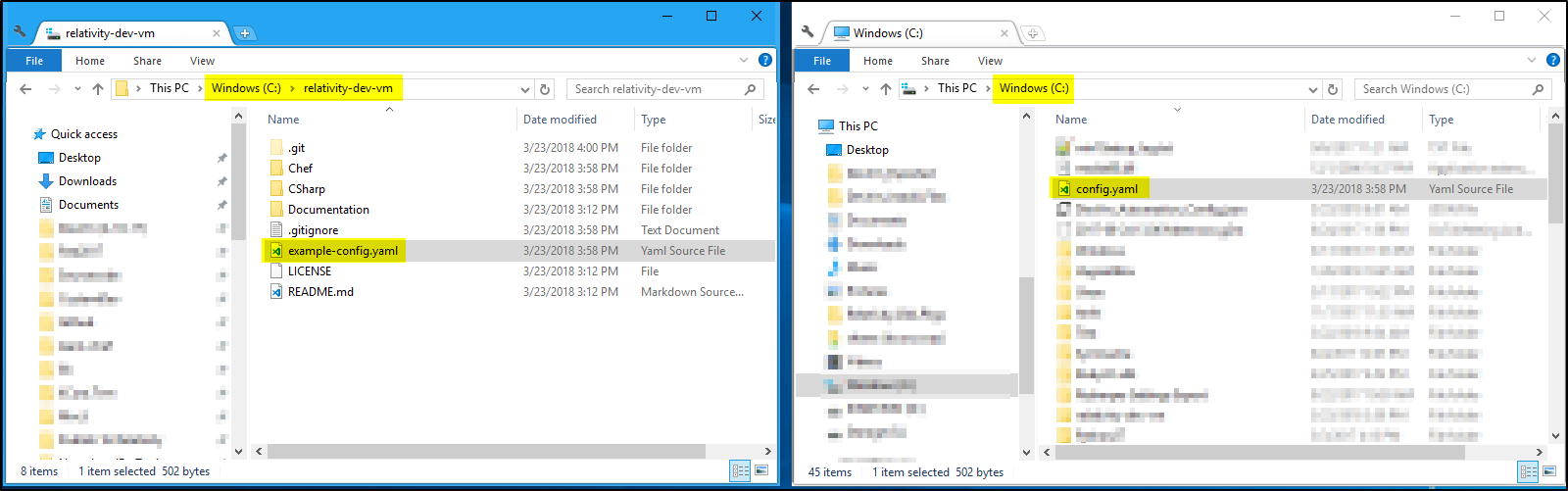
start .

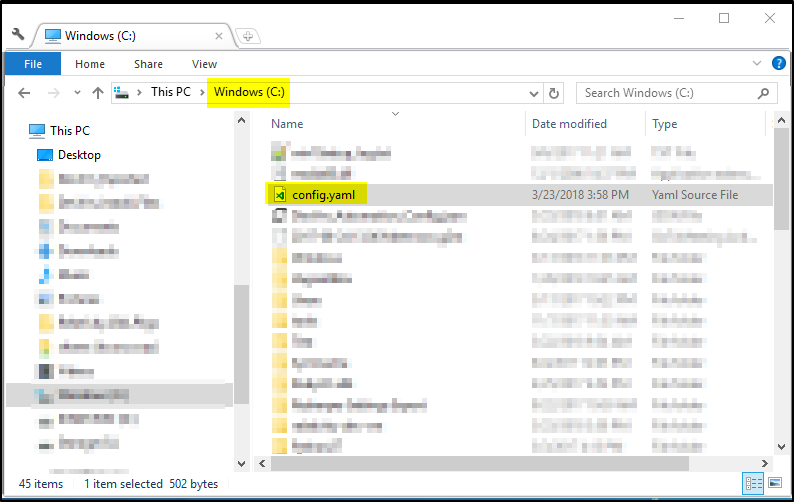


* This opens the current folder in Windows Explorer.

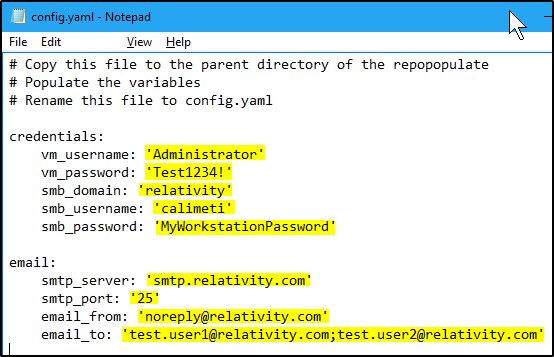


* Copy the **example-config.yaml** file from **C:\relativity-dev-vm** folder to **C drive** and rename it to **config.yaml**.





* Open the **config.yaml** file from **C drive** in Notepad.

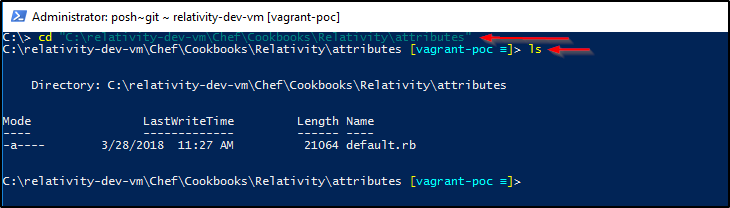


* Update the properties in the **config.yaml** file. Please find below a brief description of each property.
  + **vm\_username**: Windows Admin username of the Dev VM.
  + **vm\_password**: Windows Admin password of the Dev VM.
  + **smb\_domain**: Domain of your workstation. If your workstation is not part of a domain, use your workstation computer name.
  + **smb\_username**: Windows Admin username of your workstation.
  + **smb\_password**: Windows Admin password of your workstation.
  + **smtp\_server**, **smtp\_port**, **email\_from**, **email\_to** will be values for the email server on your domain. These values will be used to send the report of the Smoke tests run after the Dev VM creation.
* Save the changes to the **config.yaml** file.

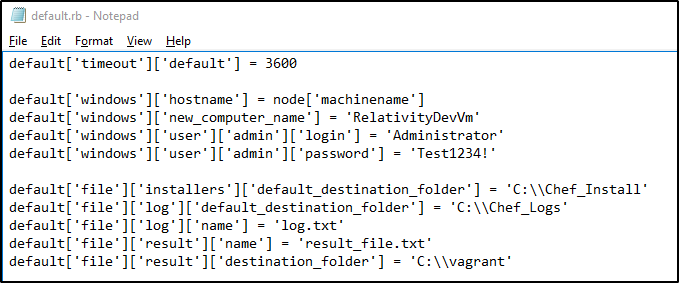
## Update Configuration file

* Navigate to the configuration file directory. Run the following command in PowerShell window.

cd "C:\relativity-dev-vm\Chef\Cookbooks\Relativity\attributes"



* The directory contains one file name **default.rb**.
* Open the **default.rb** file and update the necessary properties.

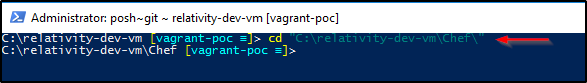


## Run the Dev VM Creation script

### Navigate to the Dev VM GIT repository

* Run the following command in PowerShell window.

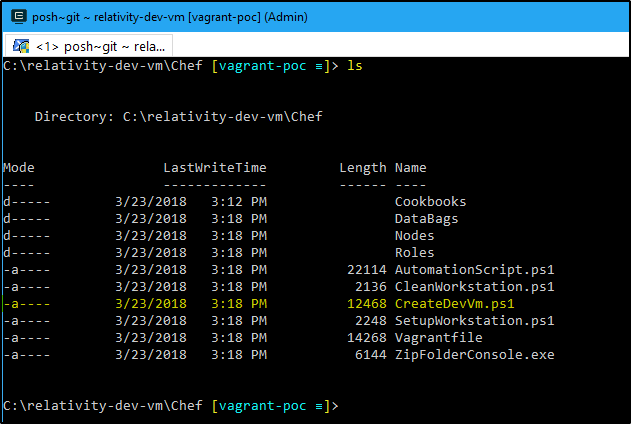
cd "C:\relativity-dev-vm\Chef\"



### List all the files in Chef folder

* Run the following command in PowerShell window.

ls

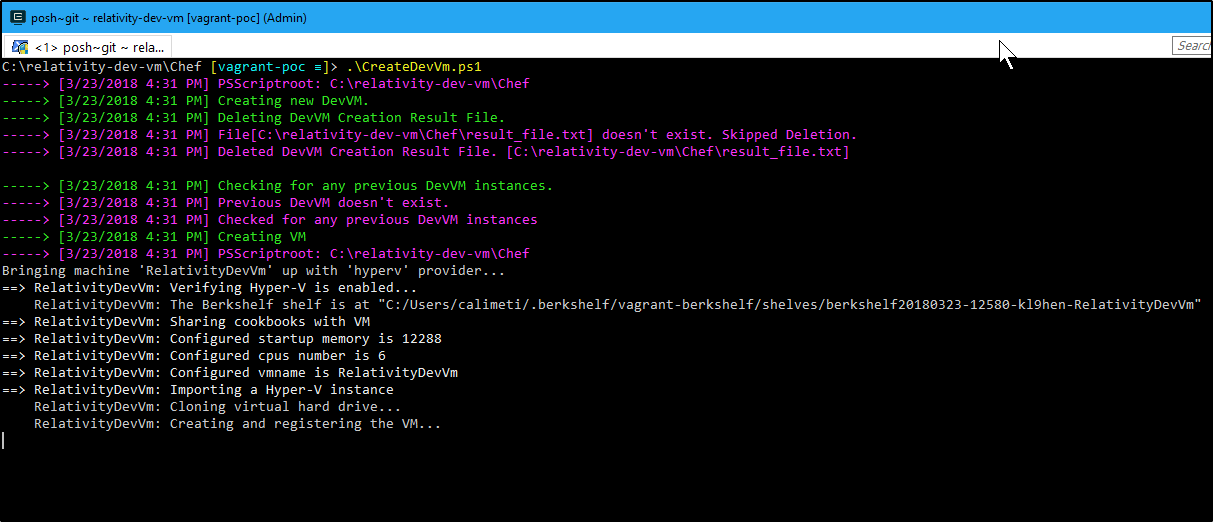


* **CreateDevVm.ps1** PowerShell script has the code to create a new Dev VM.

### Run the Creation script

* Run the following command in PowerShell window.

.\CreateDevVm.ps1



* Once the script finishes execution successfully, you should be able to see a

# Clean Workstation

* It’s important that you run the below steps in order.

## Login

* Login into your workstation with a Windows Administrator account.

## Remove Vagrant Box Image

* Run the following command in PowerShell window.

vagrant box remove "DevVmBaseImage" –force

## Uninstall Vagrant berkshelf plugin

* Run the following command in PowerShell window.

vagrant plugin uninstall vagrant-berkshelf

## Uninstall Vagrant reload plugin

* Run the following command in PowerShell window.

vagrant plugin uninstall vagrant-reload

## Uninstall Vagrant

* Run the following command in PowerShell window.

choco uninstall vagrant –force

## Uninstall Chocolatey

* Run the following command in PowerShell window.

Remove-Item -Recurse -Force "$env:ChocolateyInstall"

* Run the following command in PowerShell window.

[System.Text.RegularExpressions.Regex]::Replace([Microsoft.Win32.Registry]::CurrentUser.OpenSubKey('Environment').GetValue('PATH', '', [Microsoft.Win32.RegistryValueOptions]::DoNotExpandEnvironmentNames).ToString(), [System.Text.RegularExpressions.Regex]::Escape("$env:ChocolateyInstall\bin") + '(?>;)?', '', [System.Text.RegularExpressions.RegexOptions]::IgnoreCase) | % {[System.Environment]::SetEnvironmentVariable('PATH', $\_, 'User')}

* Run the following command in PowerShell window.

[System.Text.RegularExpressions.Regex]::Replace([Microsoft.Win32.Registry]::LocalMachine.OpenSubKey('SYSTEM\CurrentControlSet\Control\Session Manager\Environment\').GetValue('PATH', '', [Microsoft.Win32.RegistryValueOptions]::DoNotExpandEnvironmentNames).ToString(), [System.Text.RegularExpressions.Regex]::Escape("$env:ChocolateyInstall\bin") + '(?>;)?', '', [System.Text.RegularExpressions.RegexOptions]::IgnoreCase) | % {[System.Environment]::SetEnvironmentVariable('PATH', $\_, 'Machine')}

* Run the following command in PowerShell window.

if ($env:ChocolateyBinRoot -ne '' -and $env:ChocolateyBinRoot -ne $null) { Remove-Item -Recurse -Force "$env:ChocolateyBinRoot" }

* Run the following command in PowerShell window.

if ($env:ChocolateyToolsRoot -ne '' -and $env:ChocolateyToolsRoot -ne $null) { Remove-Item -Recurse -Force "$env:ChocolateyToolsRoot" }

* Run the following command in PowerShell window.

[System.Environment]::SetEnvironmentVariable("ChocolateyBinRoot", $null, 'User')

* Run the following command in PowerShell window.

[System.Environment]::SetEnvironmentVariable("ChocolateyToolsLocation", $null, 'User')

## Remove Hyper-V Virtual Switch

* Run the following command in PowerShell window.

Remove-VMSwitch "RelativityDevVmSwitch" -Force

## Uninstall Hyper-V

* Run the following command in PowerShell window.

Disable-WindowsOptionalFeature -Online -FeatureName Microsoft-Hyper-V -NoRestart

* **Restart your workstation.**

## Uninstall Chef Development Kit

* Open **Programs and Features** application.
* Search for **Chef Development Kit**, select it and click Uninstall.

