1. Vagrant + Virtualbox Setup on Windows.

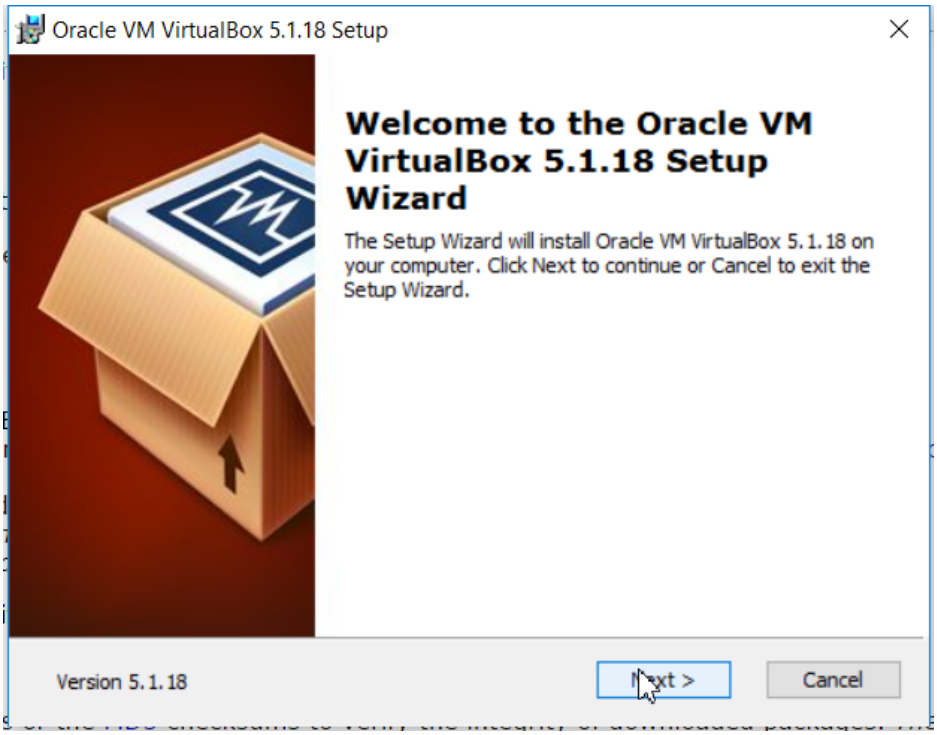
We will need three tools listed below.

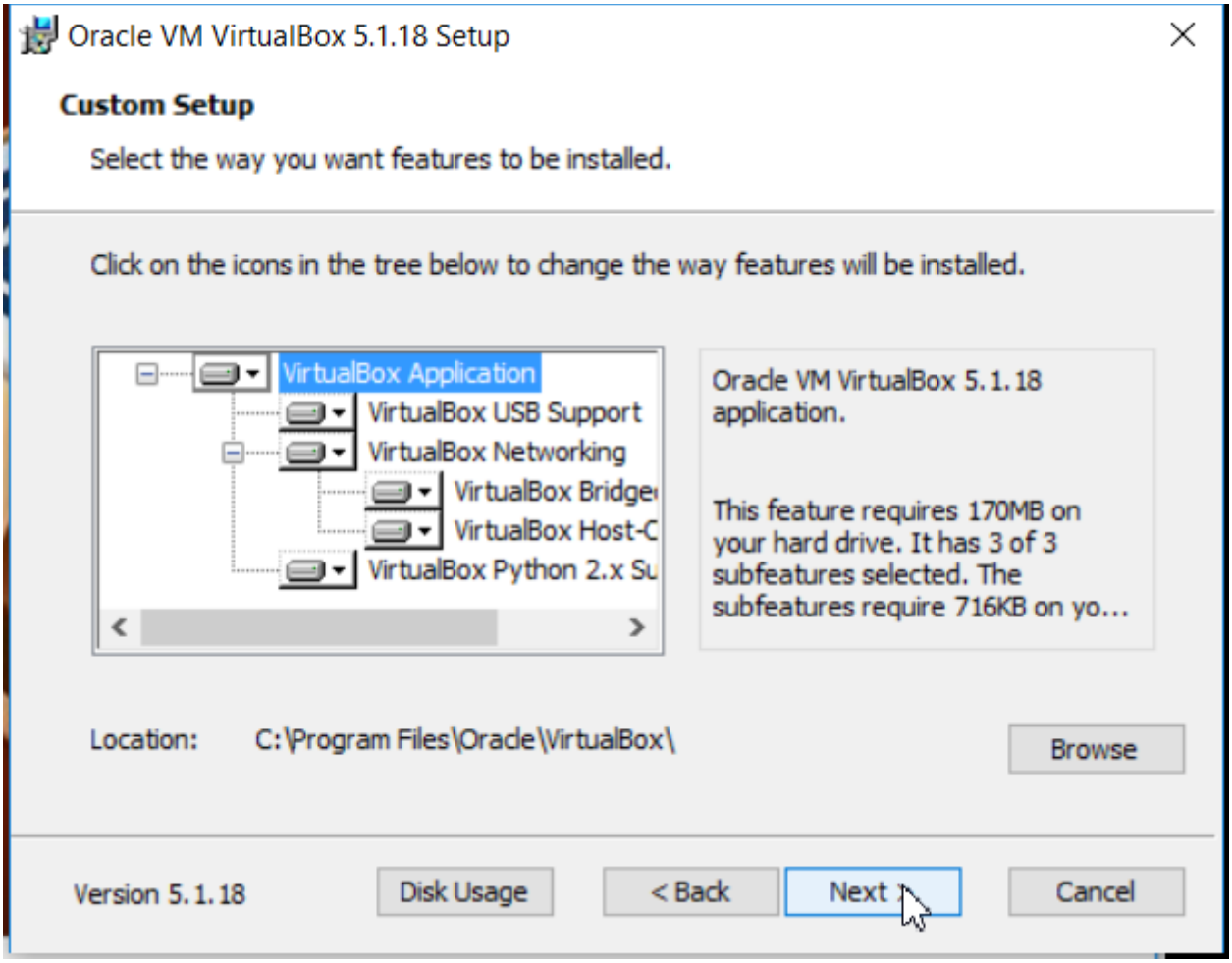
- Virtualbox
- Vagrant
- Git Bash

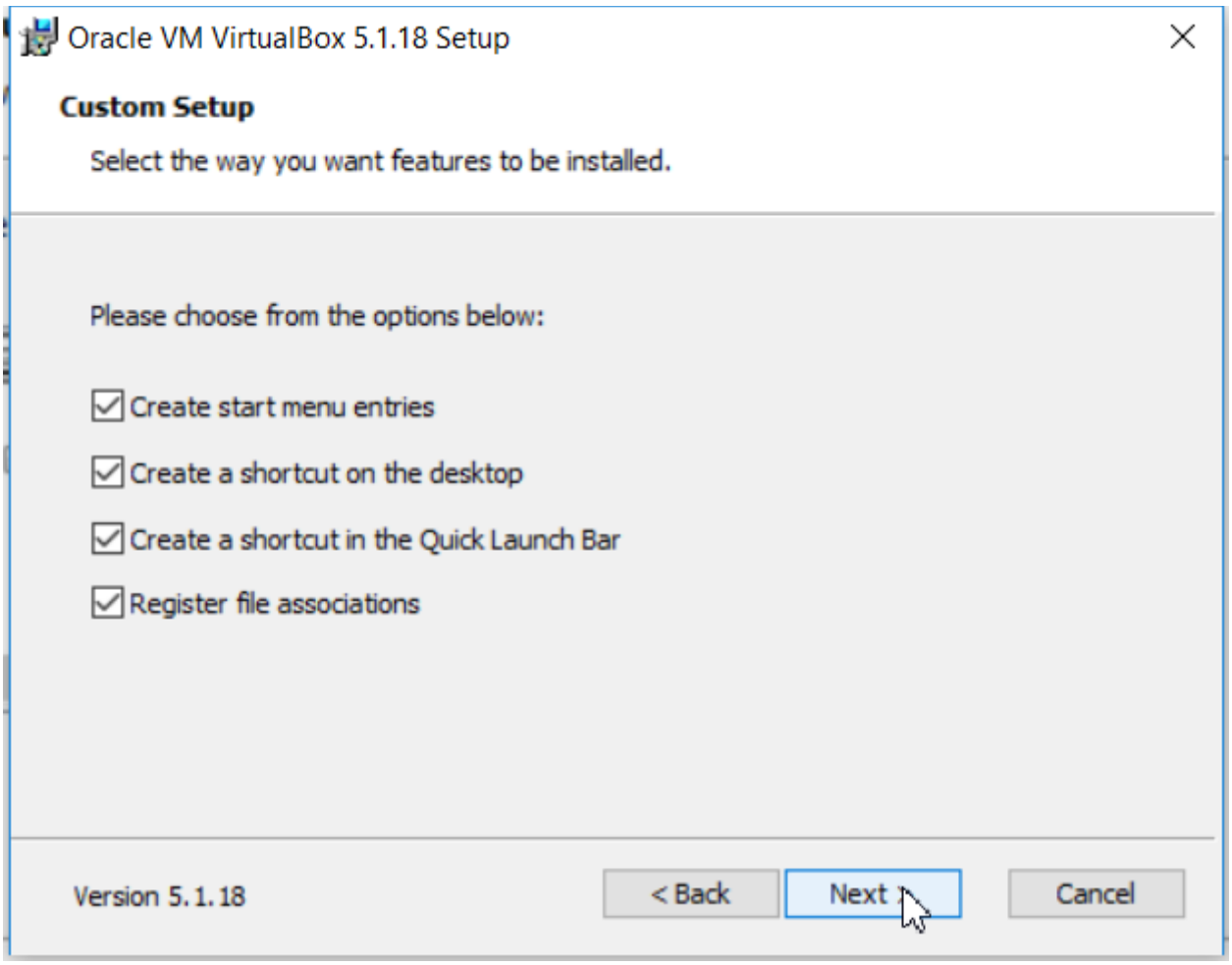
Installing virtualbox

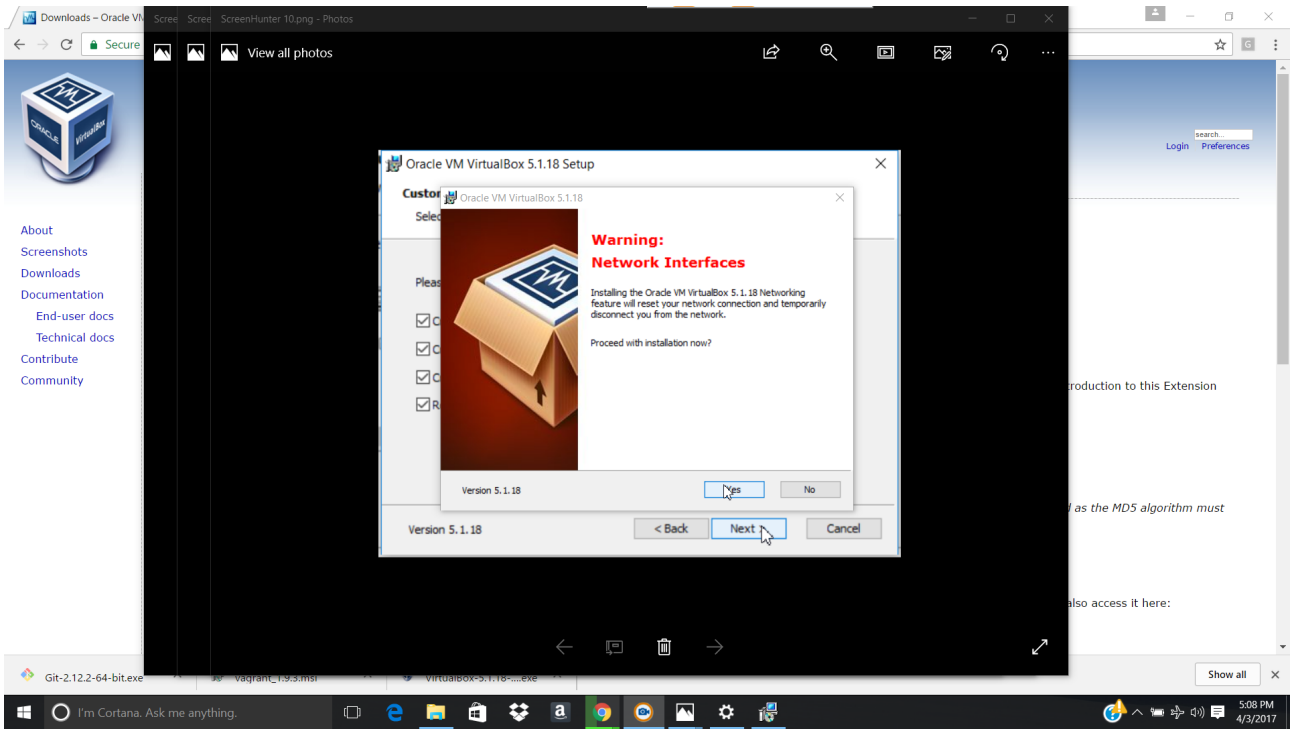
➔ Go to Virtualbox Download page, Click Windows hosts.

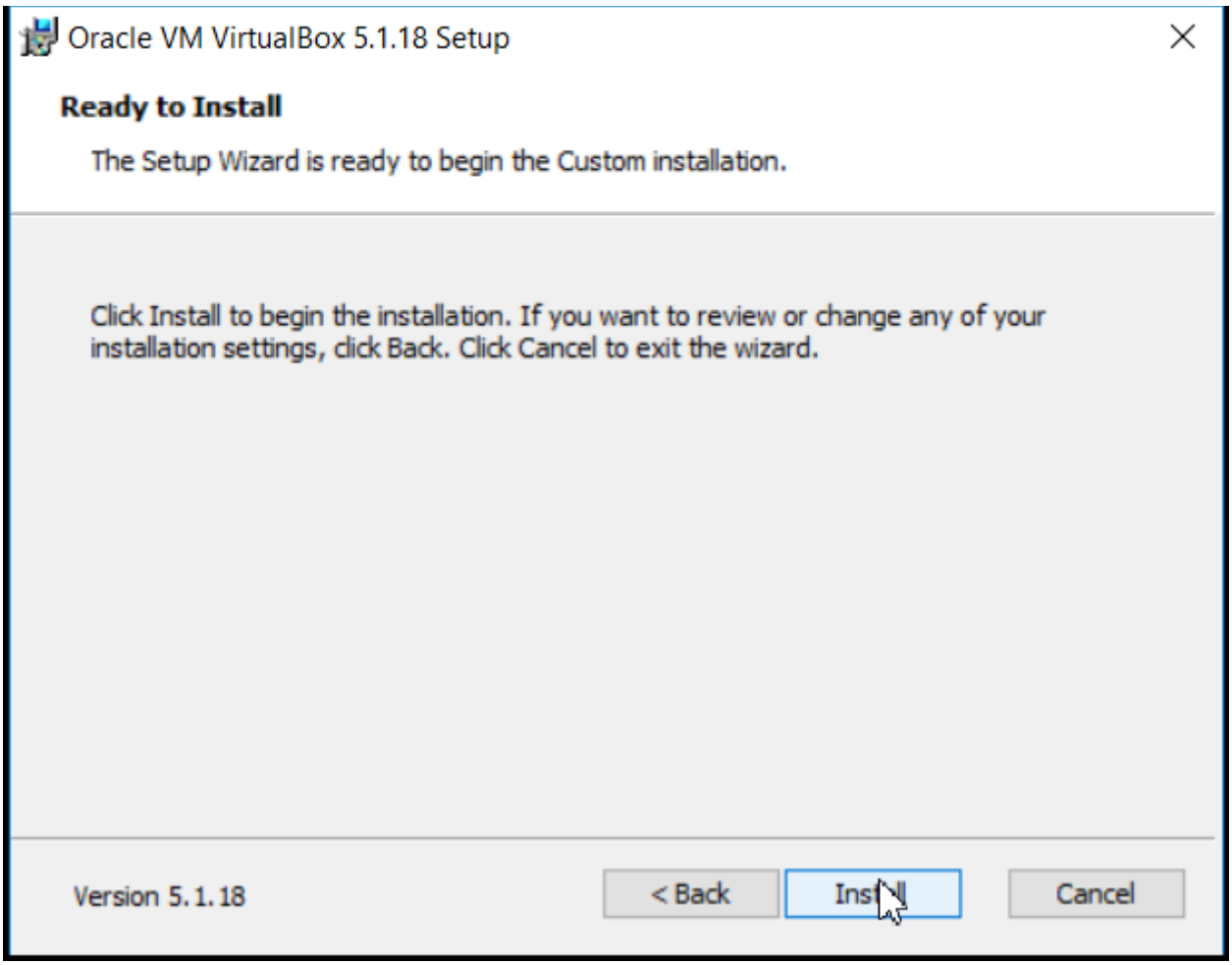
➔ Open VirtualBox exe file and follow below screenshots.

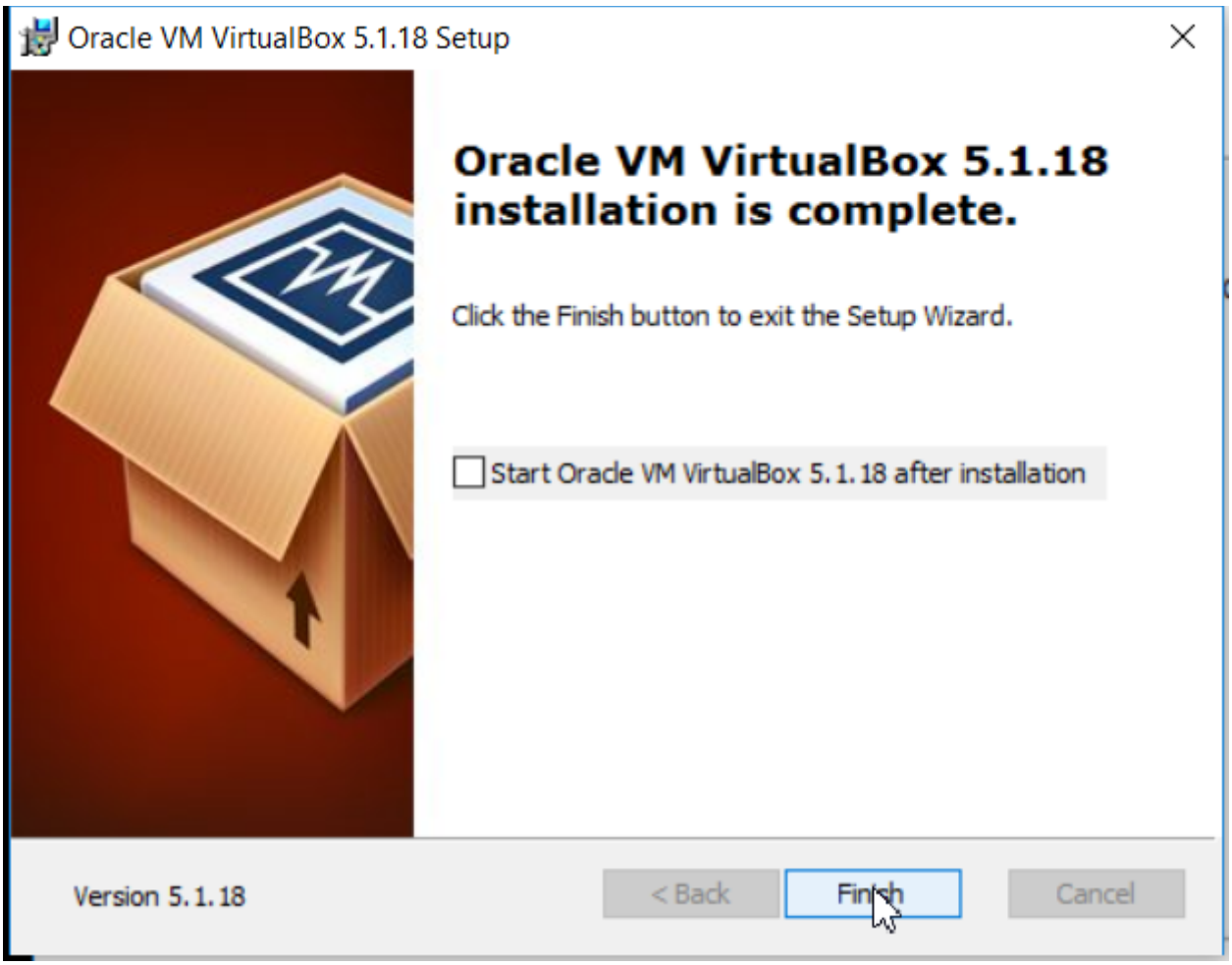



Oracle VM VirtualBox 5.1.18 Setup
Custom Setup
Select the way you want features to be installed.
Click on the icons in the tree below to change the way features will be installed.
VirtualBox Application
VirtualBox USB Support
VirtualBox Networking
VirtualBox Bridge
VirtualBox Host-C
VirtualBox Python 2.x Su
Oracle VM VirtualBox 5.1.18 application.
This feature requires 170MB on your hard drive. It has 3 of 3 subfeatures selected. The subfeatures require 716KB on yo...
Location:
C:\Program Files\Oracle\VirtualBox\
Browse
Version 5.1.18
Disk Usage
< Back
Next >
Cancel


Oracle VM VirtualBox 5.1.18 Setup
Custom Setup
Select the way you want features to be installed.
Please choose from the options below:
Create start menu entries
Create a shortcut on the desktop
Create a shortcut in the Quick Launch Bar
Register file associations
Version 5.1.18
< Back
Next >
Cancel

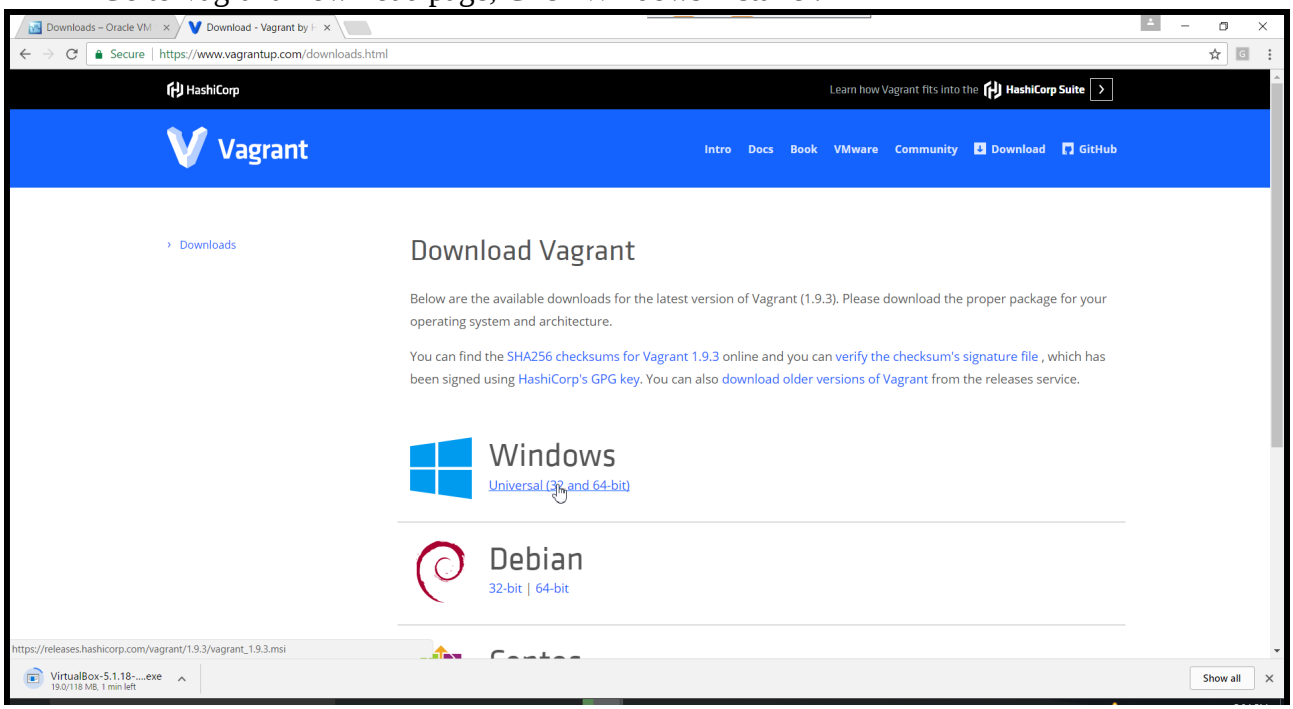

View all photos
Oracle VM VirtualBox 5.1.18 Setup
Warning:
Network Interfaces
Installing the Oracle VM VirtualBox 5.1.18 Networking feature will reset your network connection and temporarily disconnect you from the network.
Proceed with installation now?
Version 5.1.18
Yes
No
< Back
Next >
Cancel
About
Screenshots
Downloads
Documentation
End-user docs
Technical docs
Contribute
Community
Git-2.12.2-64-bit.exe
I'm Cortana. Ask me anything.
5:08 PM
4/3/2017


Oracle VM VirtualBox 5.1.18 Setup
Ready to Install
The Setup Wizard is ready to begin the Custom installation.
Click Install to begin the installation. If you want to review or change any of your installation settings, click Back. Click Cancel to exit the wizard.
Version 5.1.18
< Back
Cancel

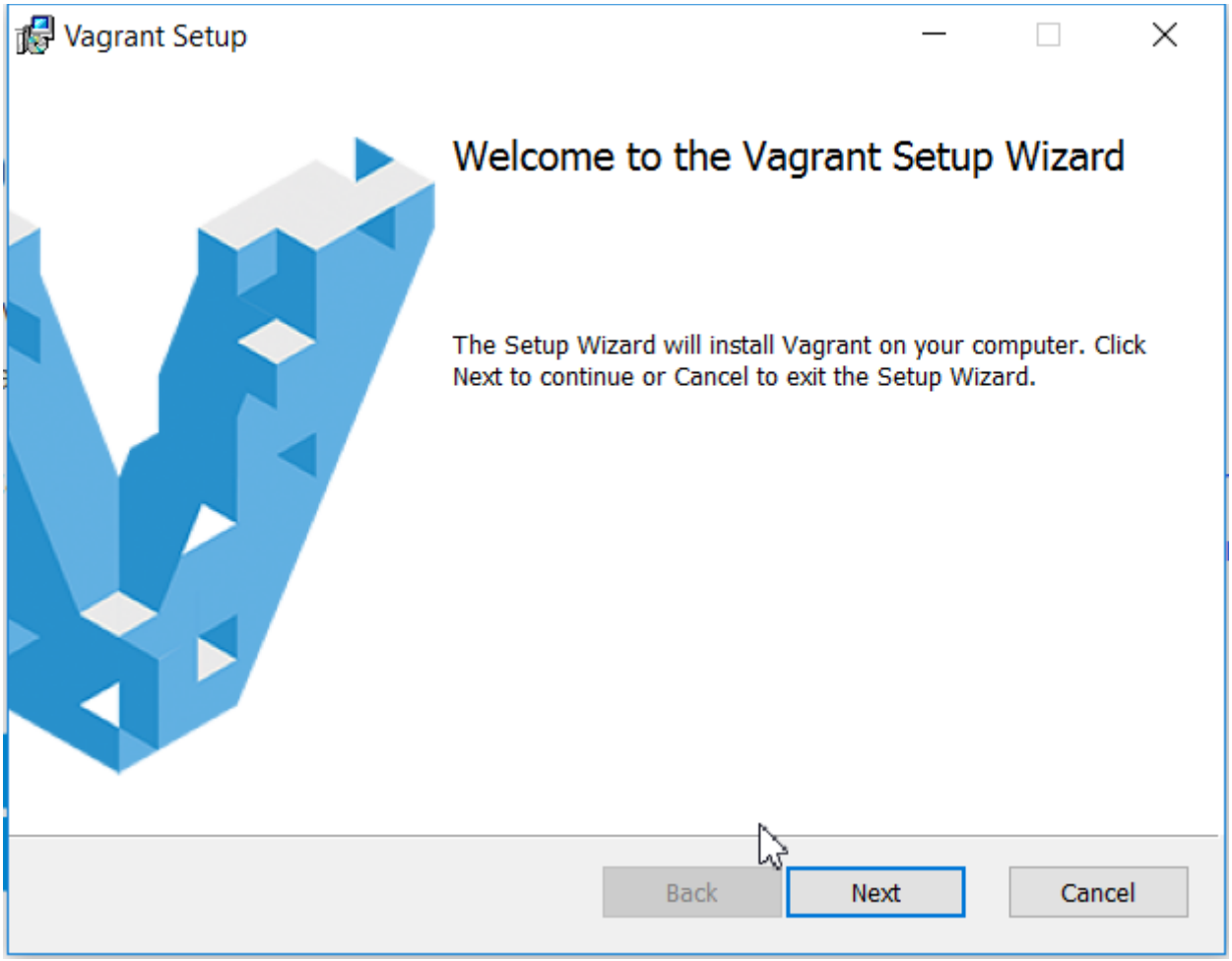


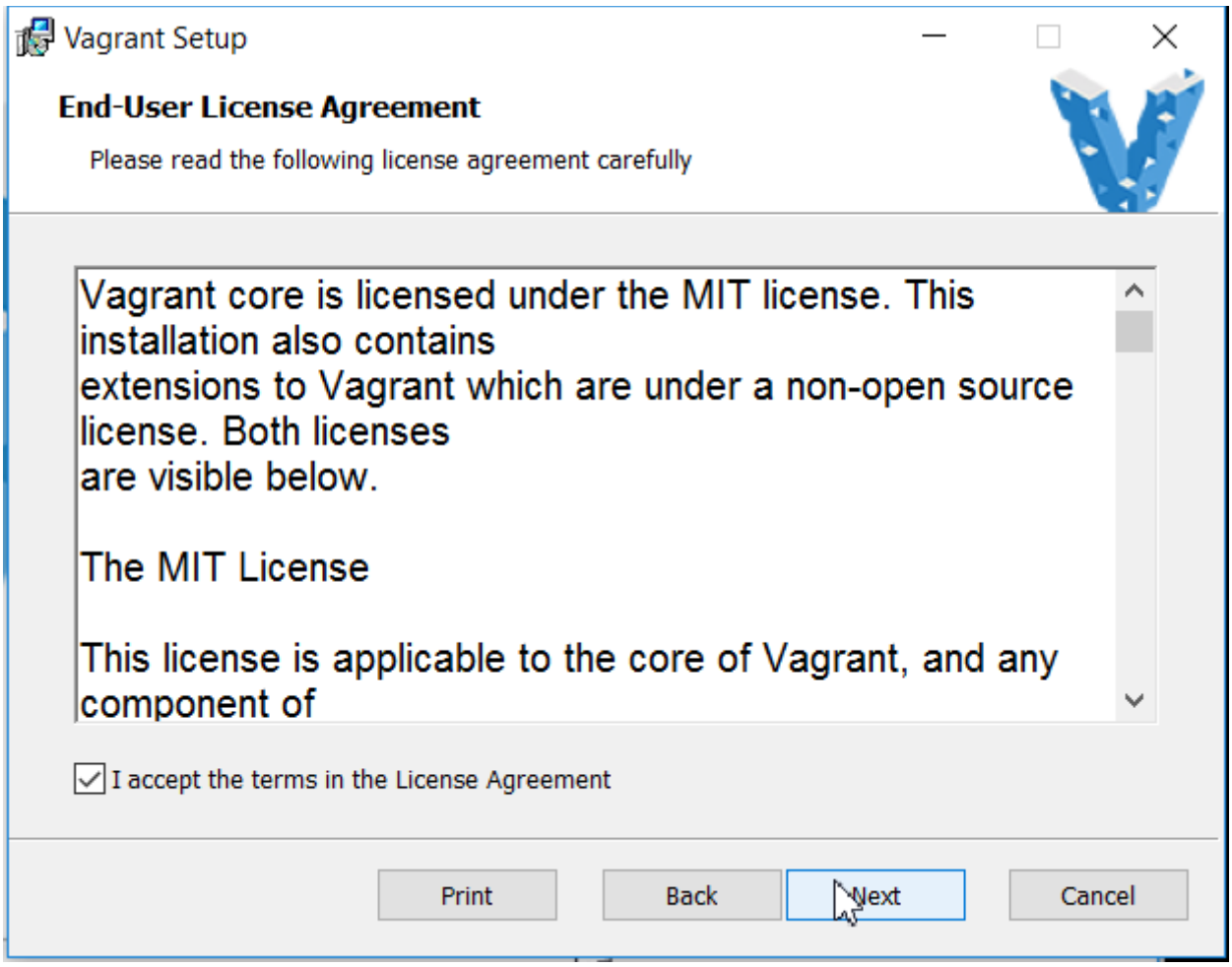
Download Vagrant.

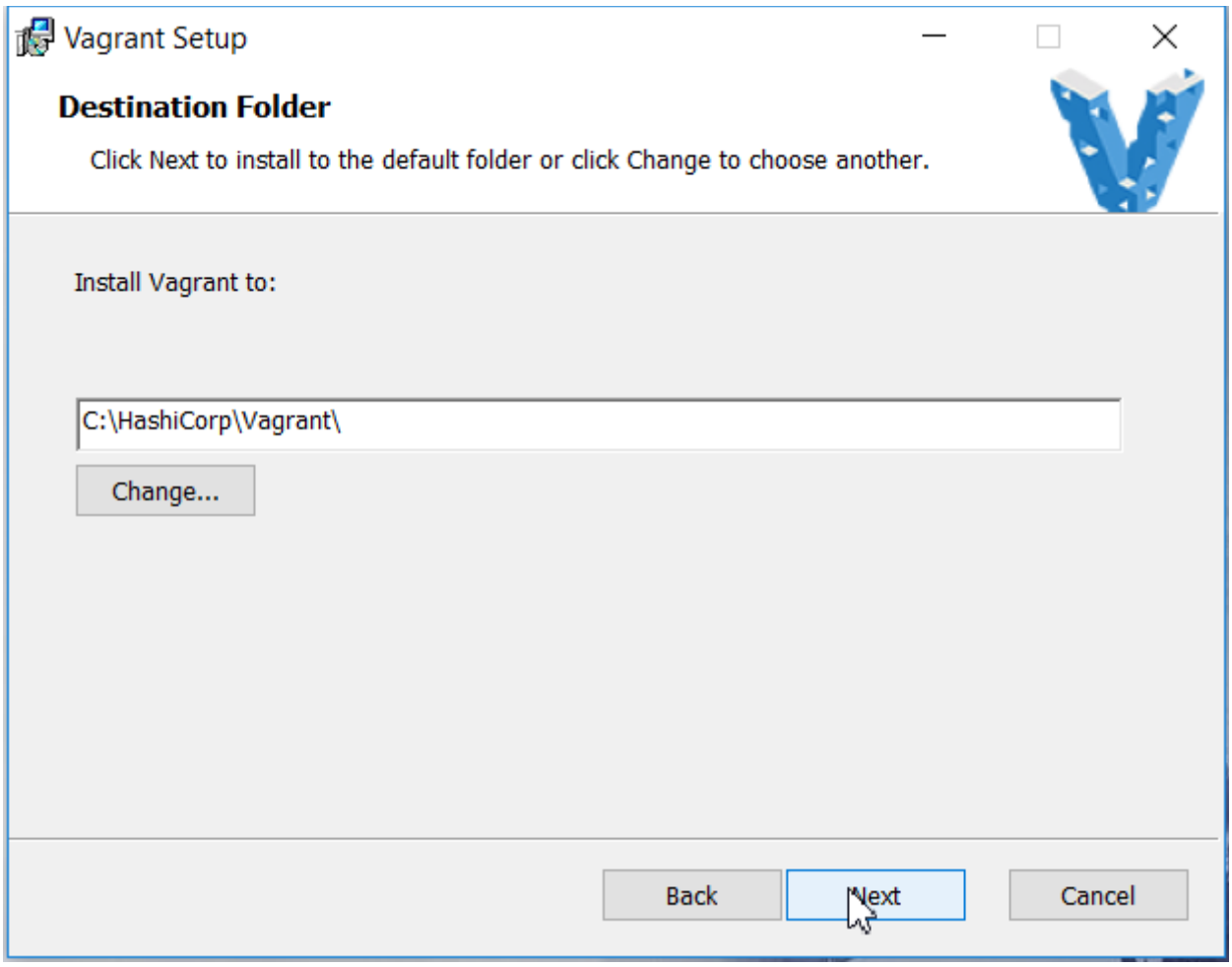
➔ Go to Vagrant Download page, Click Windows Installer.

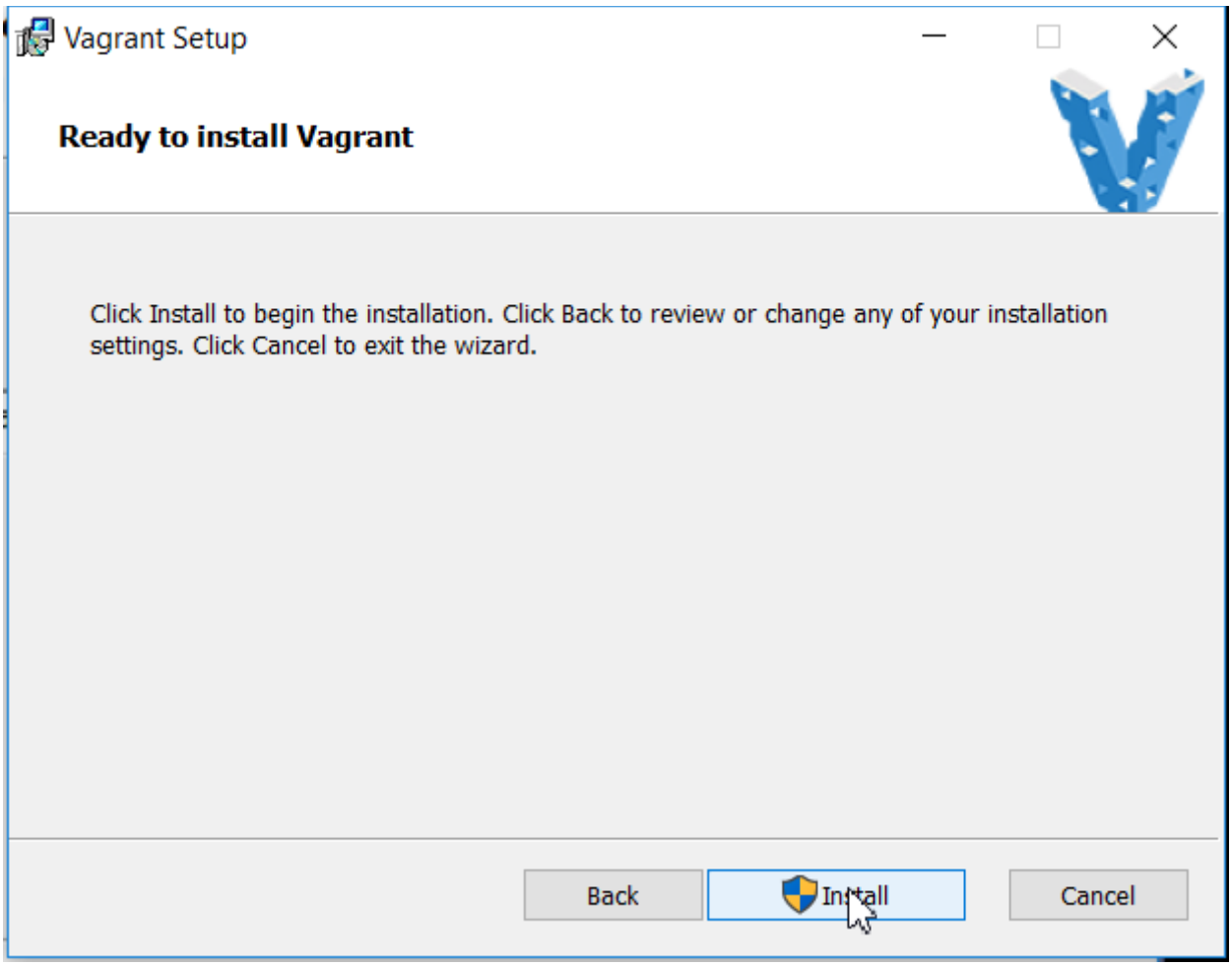


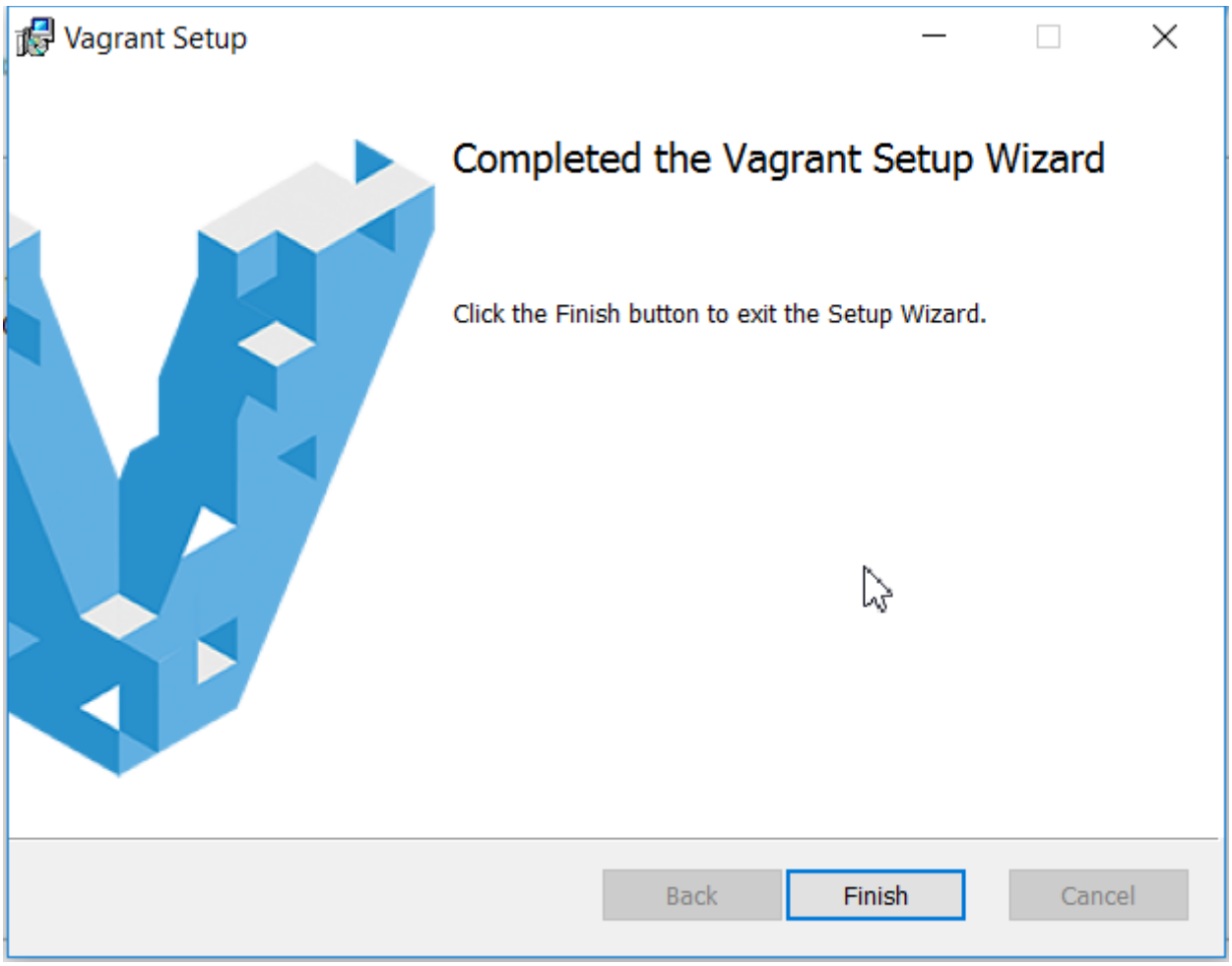
➔ Open vagrant msi file and follow the screenshot.




Vagrant Setup
End-User License Agreement
Please read the following license agreement carefully
Vagrant core is licensed under the MIT license. This installation also contains
extensions to Vagrant which are under a non-open source license. Both licenses
are visible below.
The MIT License
This license is applicable to the core of Vagrant, and any component of
I accept the terms in the License Agreement
Print
Back
Next
Cancel

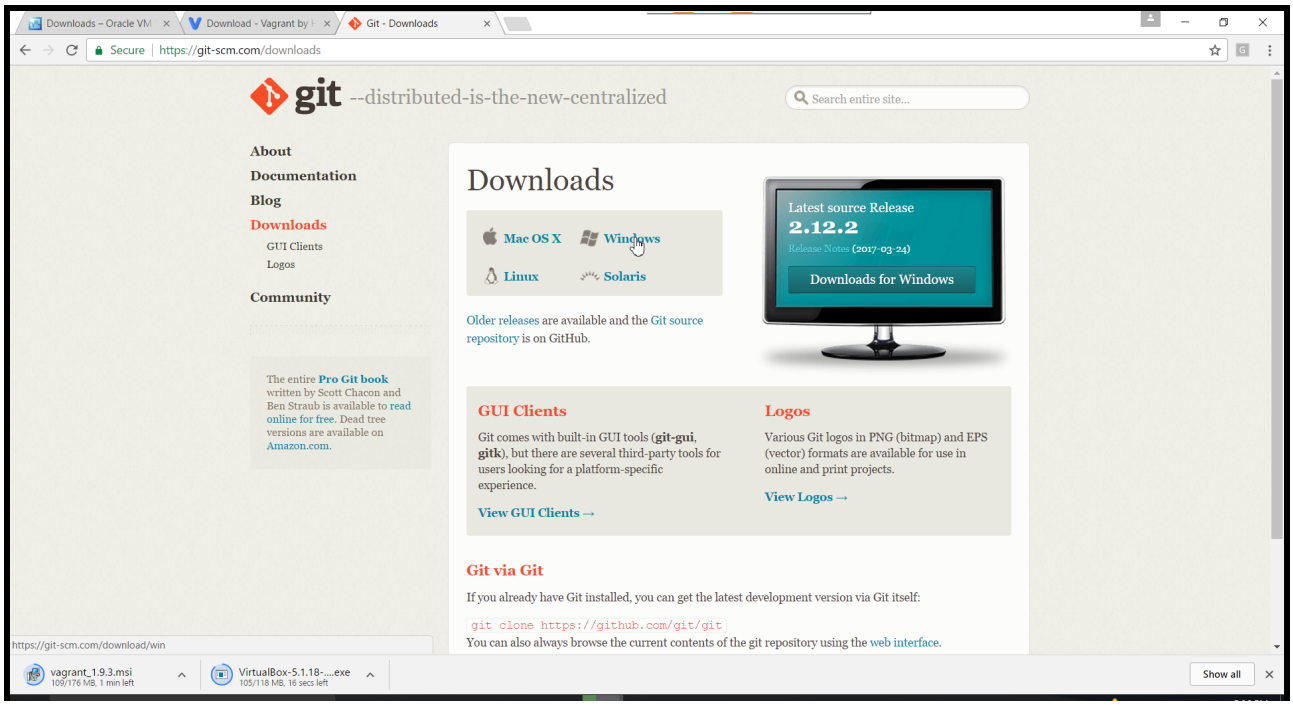

Vagrant Setup
Destination Folder
Click Next to install to the default folder or click Change to choose another.
Install Vagrant to:
C:\HashiCorp\Vagrant\
Change...
Back
Next
Cancel


Vagrant Setup
Ready to install Vagrant
Click Install to begin the installation. Click Back to review or change any of your installation settings. Click Cancel to exit the wizard.
Back
Install
Cancel

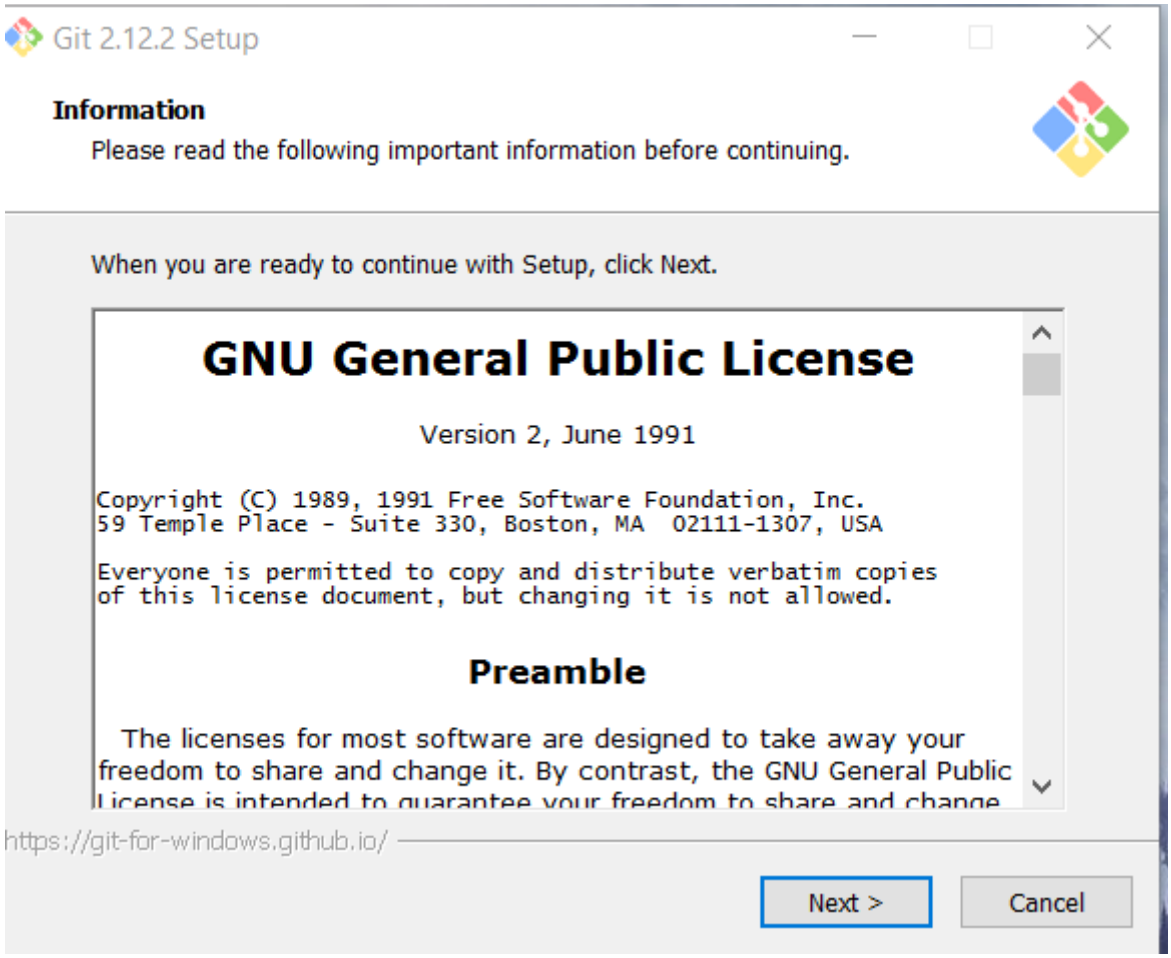


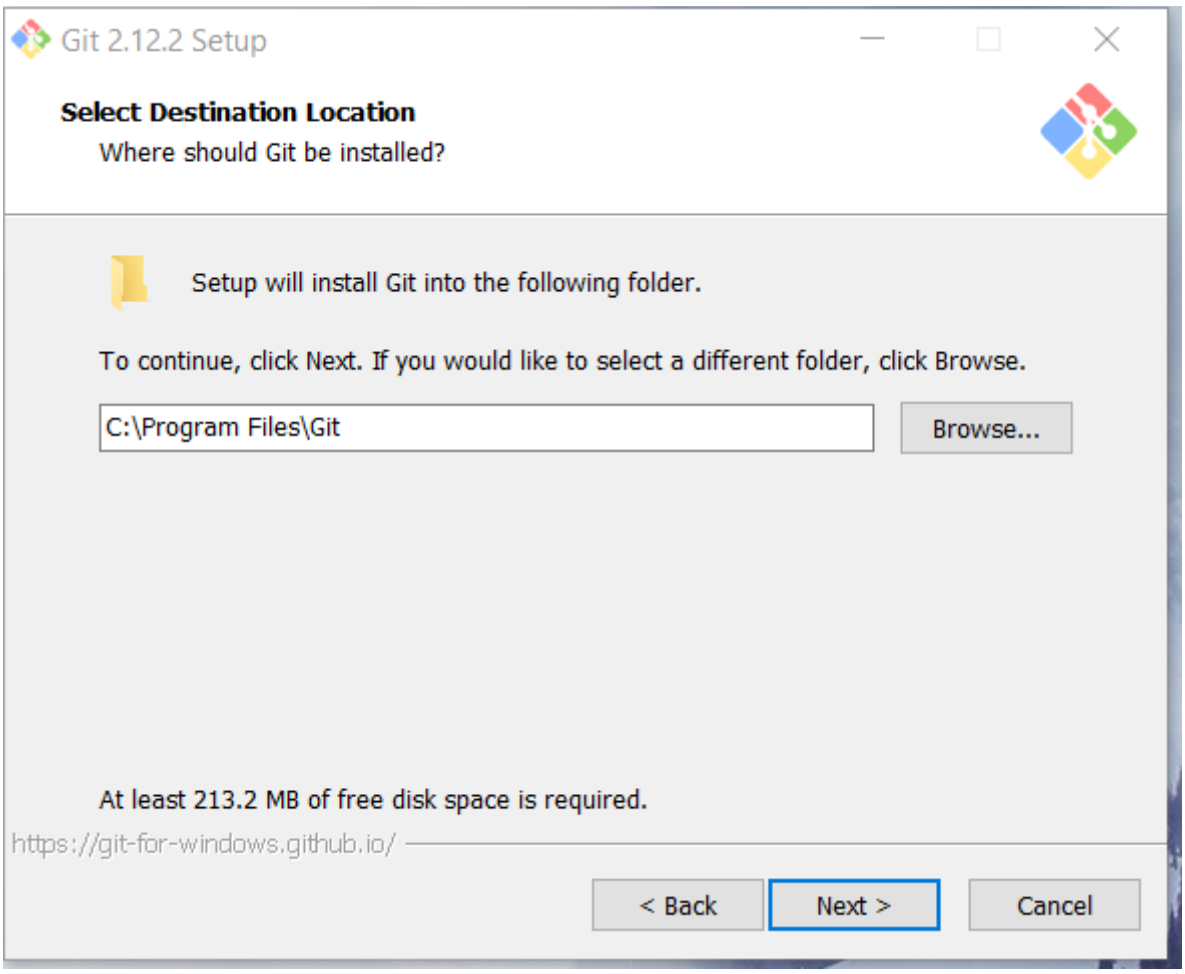
Download git for windows.

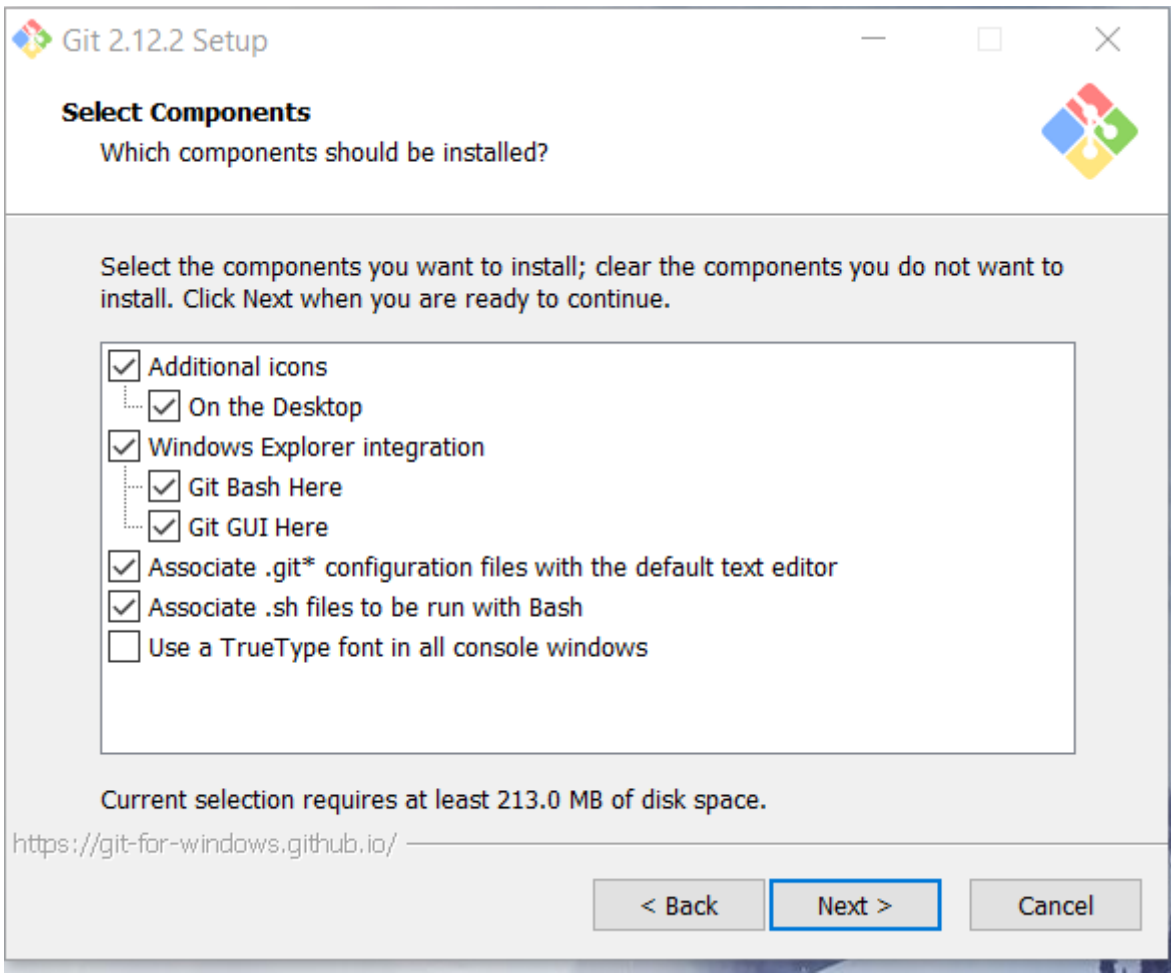
➔ Go to git scm download page, Select windows.

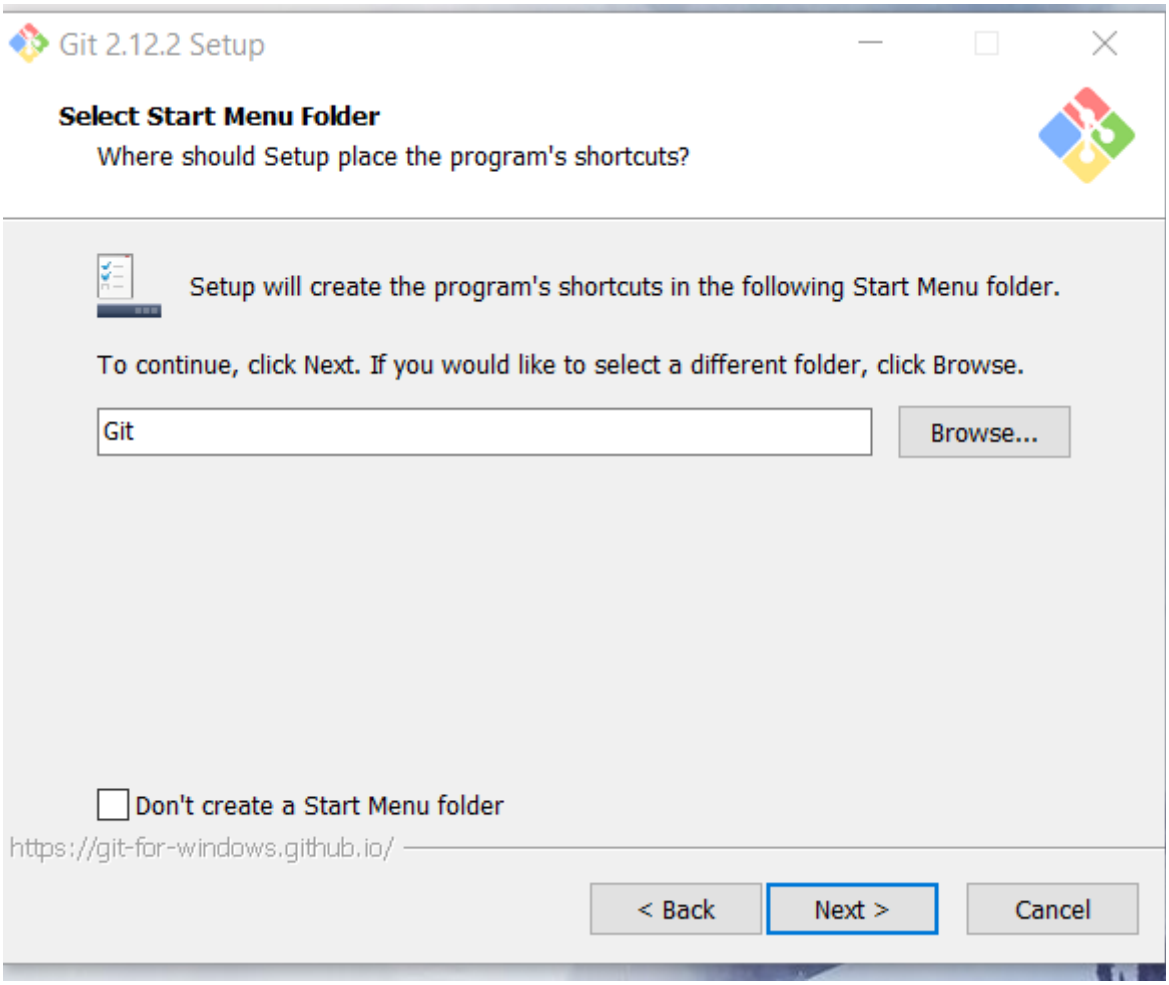


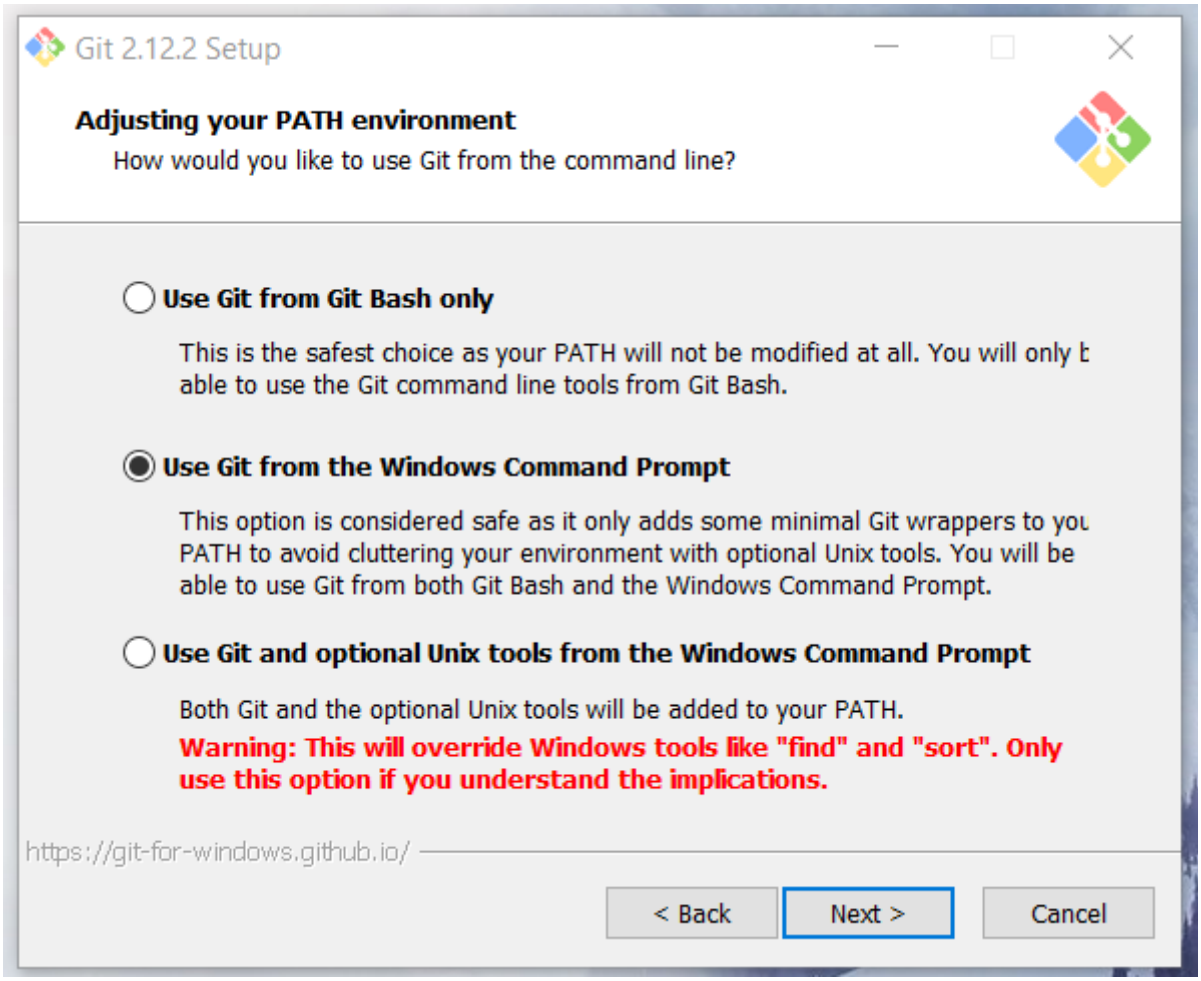
➔ Open git installable and follow the screenshots below.

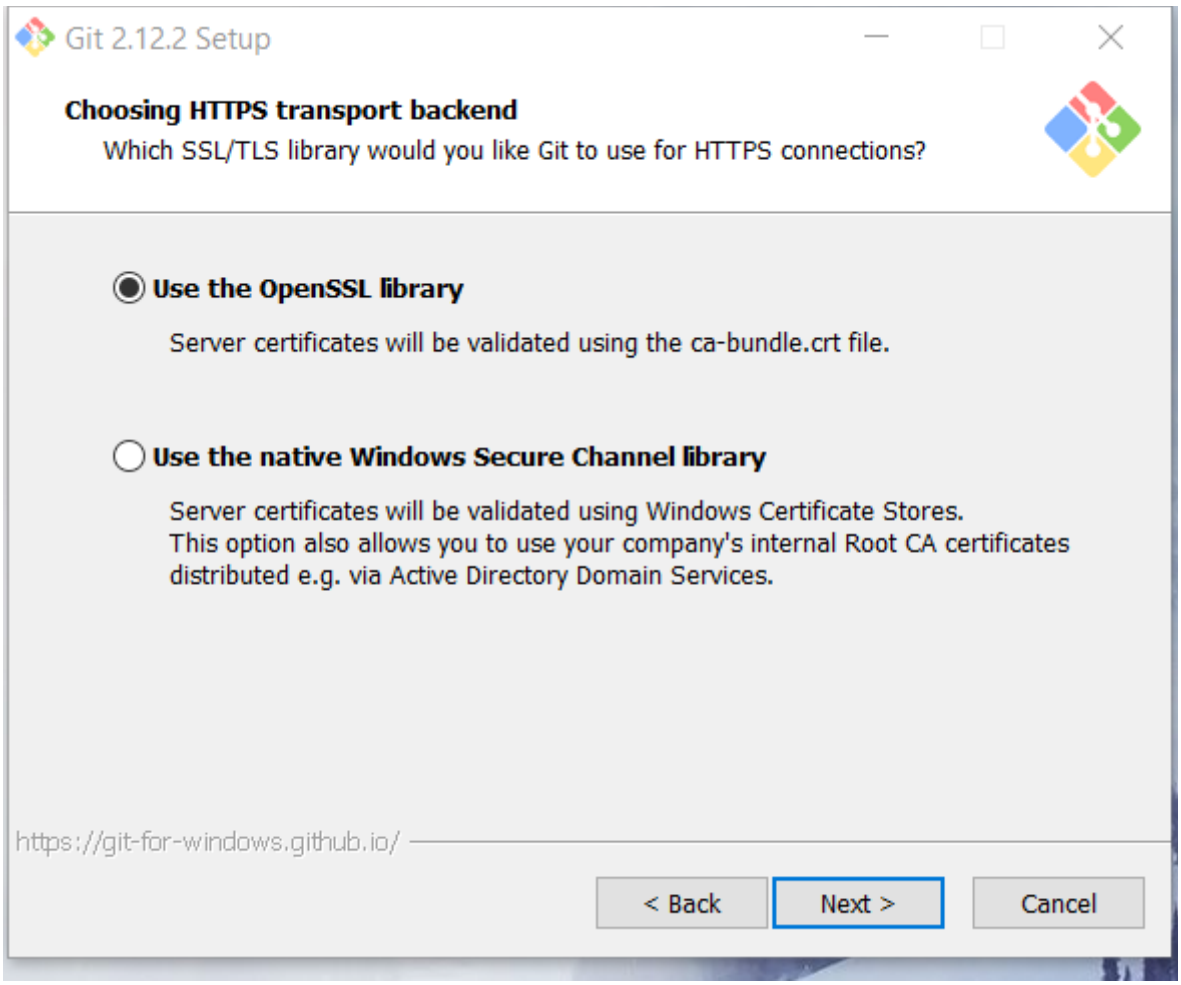


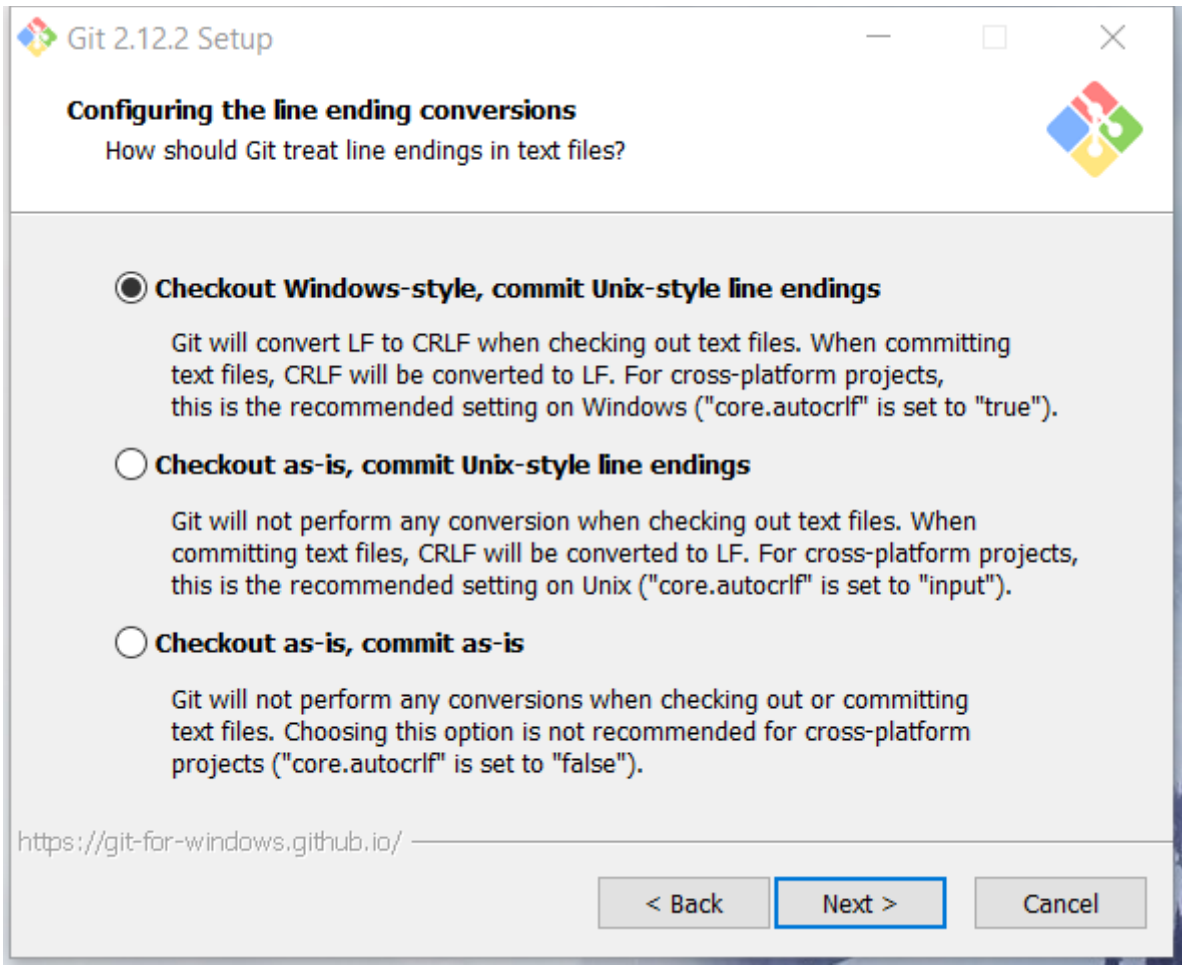

Git 2.12.2 Setup
Select Destination Location
Where should Git be installed?
Setup will install Git into the following folder.
To continue, click Next. If you would like to select a different folder, click Browse.
C:\Program Files\Git
Browse...
At least 213.2 MB of free disk space is required.
https://git-for-windows.github.io/
< Back
Next >
Cancel

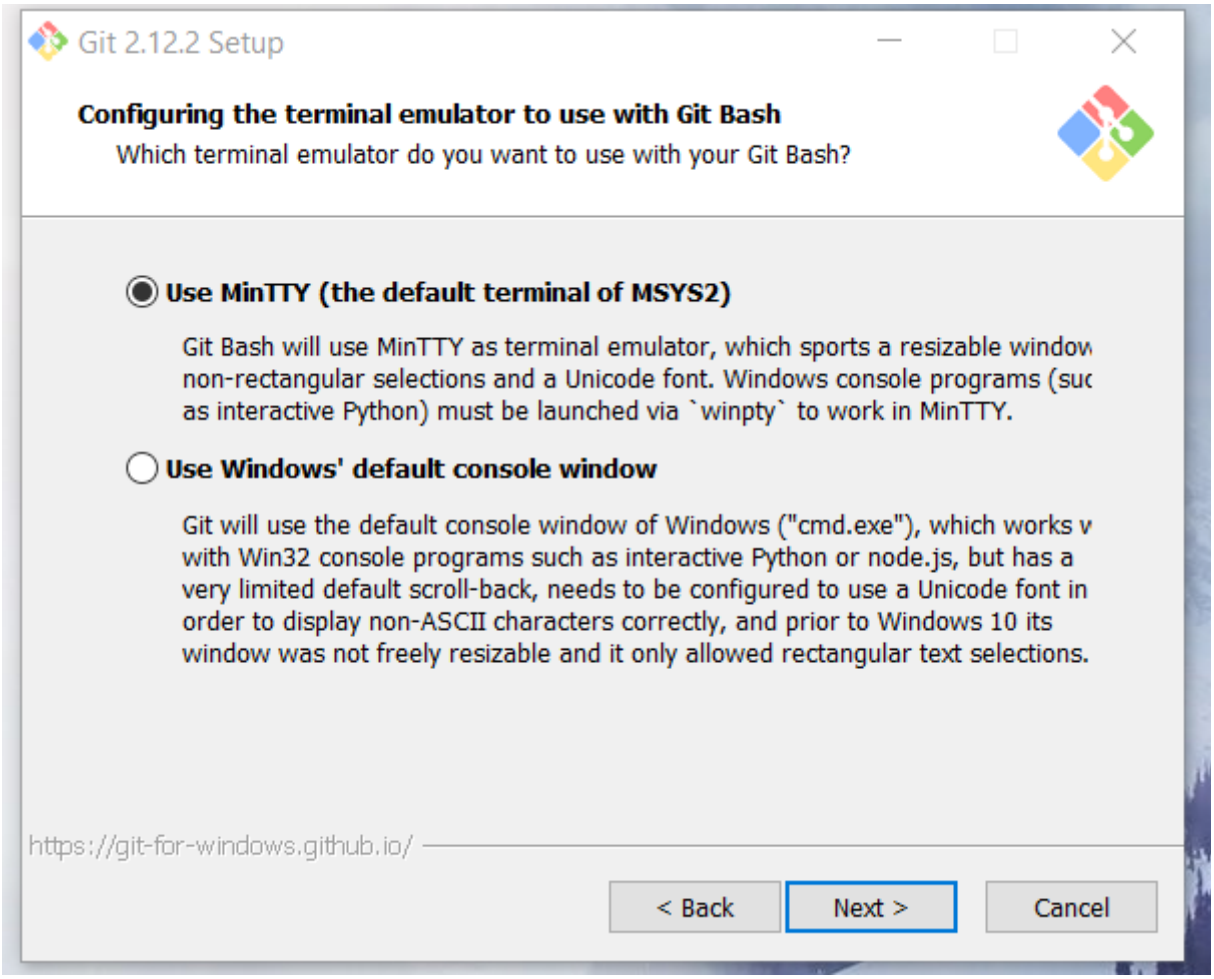

Git 2.12.2 Setup
Select Components
Which components should be installed?
Select the components you want to install; clear the components you do not want to install. Click Next when you are ready to continue.
Additional icons
On the Desktop
Windows Explorer integration
Git Bash Here
Git GUI Here
Associate .git* configuration files with the default text editor
Associate .sh files to be run with Bash
Use a TrueType font in all console windows
Current selection requires at least 213.0 MB of disk space.
https://git-for-windows.github.io/
< Back
Next >
Cancel

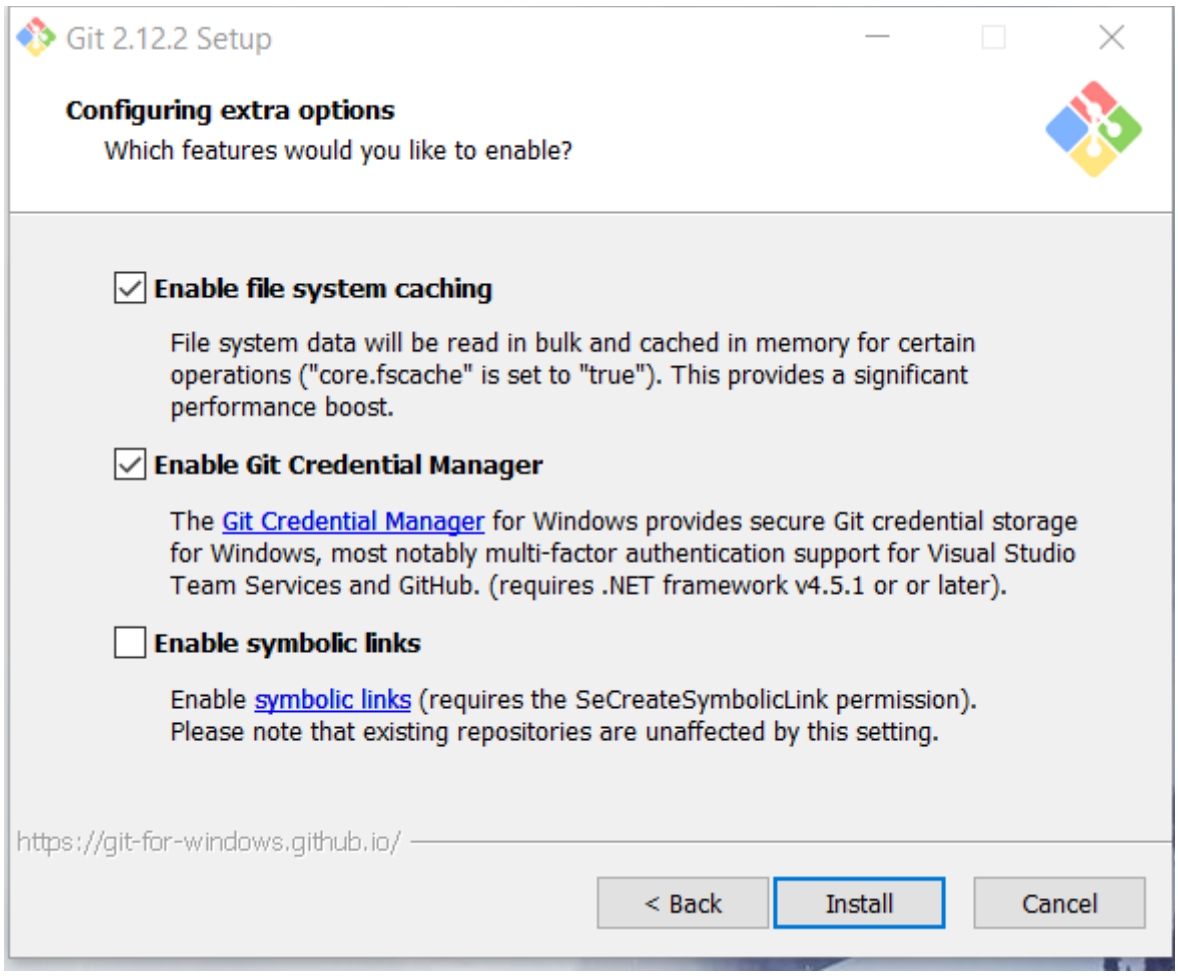

Git 2.12.2 Setup
Select Start Menu Folder
Where should Setup place the program's shortcuts?
Setup will create the program's shortcuts in the following Start Menu folder.
To continue, click Next. If you would like to select a different folder, click Browse.
Git
Browse...
Don't create a Start Menu folder
https://git-for-windows.github.io/
< Back
Next >
Cancel


Git 2.12.2 Setup
Adjusting your PATH environment
How would you like to use Git from the command line?
Use Git from Git Bash only
This is the safest choice as your PATH will not be modified at all. You will only b
able to use the Git command line tools from Git Bash.
Use Git from the Windows Command Prompt
This option is considered safe as it only adds some minimal Git wrappers to you
PATH to avoid cluttering your environment with optional Unix tools. You will be
able to use Git from both Git Bash and the Windows Command Prompt.
Use Git and optional Unix tools from the Windows Command Prompt
Both Git and the optional Unix tools will be added to your PATH.
Warning: This will override Windows tools like "find" and "sort". Only
use this option if you understand the implications.
https://git-for-windows.github.io/
< Back
Next >
Cancel


Git 2.12.2 Setup
Choosing HTTPS transport backend
Which SSL/TLS library would you like Git to use for HTTPS connections?
Use the OpenSSL library
Server certificates will be validated using the ca-bundle.crt file.
Use the native Windows Secure Channel library
Server certificates will be validated using Windows Certificate Stores.
This option also allows you to use your company's internal Root CA certificates
distributed e.g. via Active Directory Domain Services.
https://git-for-windows.github.io/
< Back
Next >
Cancel


Git 2.12.2 Setup
Configuring the line ending conversions
How should Git treat line endings in text files?
Checkout Windows-style, commit Unix-style line endings
Git will convert LF to CRLF when checking out text files. When committing text files, CRLF will be converted to LF. For cross-platform projects, this is the recommended setting on Windows ("core.autocrlf" is set to "true").
Checkout as-is, commit Unix-style line endings
Git will not perform any conversion when checking out text files. When committing text files, CRLF will be converted to LF. For cross-platform projects, this is the recommended setting on Unix ("core.autocrlf" is set to "input").
Checkout as-is, commit as-is
Git will not perform any conversions when checking out or committing text files. Choosing this option is not recommended for cross-platform projects ("core.autocrlf" is set to "false").
https://git-for-windows.github.io/
< Back
Next >
Cancel

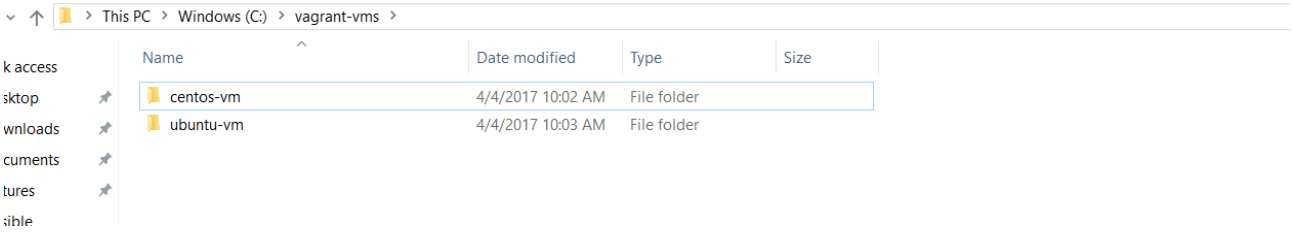

Git 2.12.2 Setup
Configuring the terminal emulator to use with Git Bash
Which terminal emulator do you want to use with your Git Bash?
Use MinTTY (the default terminal of MSYS2)
Git Bash will use MinTTY as terminal emulator, which sports a resizable window
non-rectangular selections and a Unicode font. Windows console programs (suc
as interactive Python) must be launched via `winpty` to work in MinTTY.
Use Windows' default console window
Git will use the default console window of Windows ("cmd.exe"), which works v
with Win32 console programs such as interactive Python or node.js, but has a
very limited default scroll-back, needs to be configured to use a Unicode font in
order to display non-ASCII characters correctly, and prior to Windows 10 its
window was not freely resizable and it only allowed rectangular text selections.
https://git-for-windows.github.io/
< Back
Next >
Cancel


Git 2.12.2 Setup
Configuring extra options
Which features would you like to enable?
Enable file system caching
File system data will be read in bulk and cached in memory for certain operations ("core.fscache" is set to "true"). This provides a significant performance boost.
Enable Git Credential Manager
The Git Credential Manager for Windows provides secure Git credential storage for Windows, most notably multi-factor authentication support for Visual Studio Team Services and GitHub. (requires .NET framework v4.5.1 or or later).
Enable symbolic links
Enable symbolic links (requires the SeCreateSymbolicLink permission). Please note that existing repositories are unaffected by this setting.
https://git-for-windows.github.io/
< Back
Install
Cancel

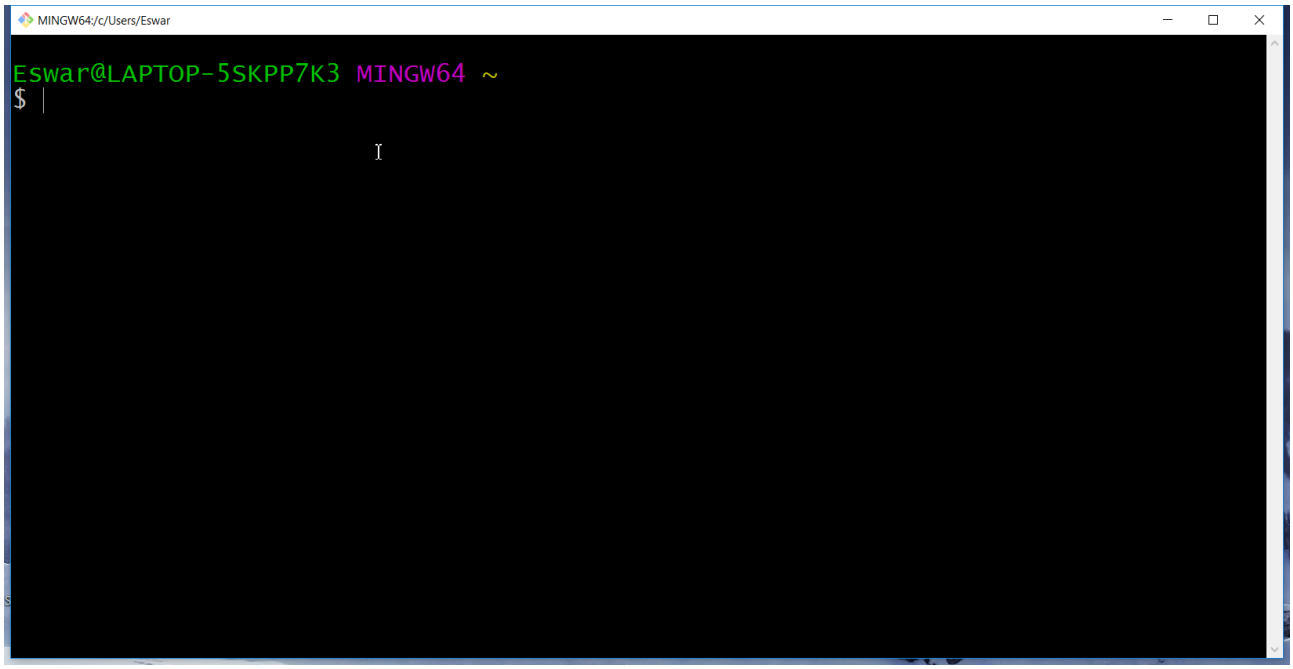
2. Creating Centos & Ubuntu VM's using Vagrant tool.

➔ **Go to C drive => Create folder named vagrant-vms**

➔ **Go to C:\vagrant-vms => Create two folders named centos-vm & ubuntu-vm**

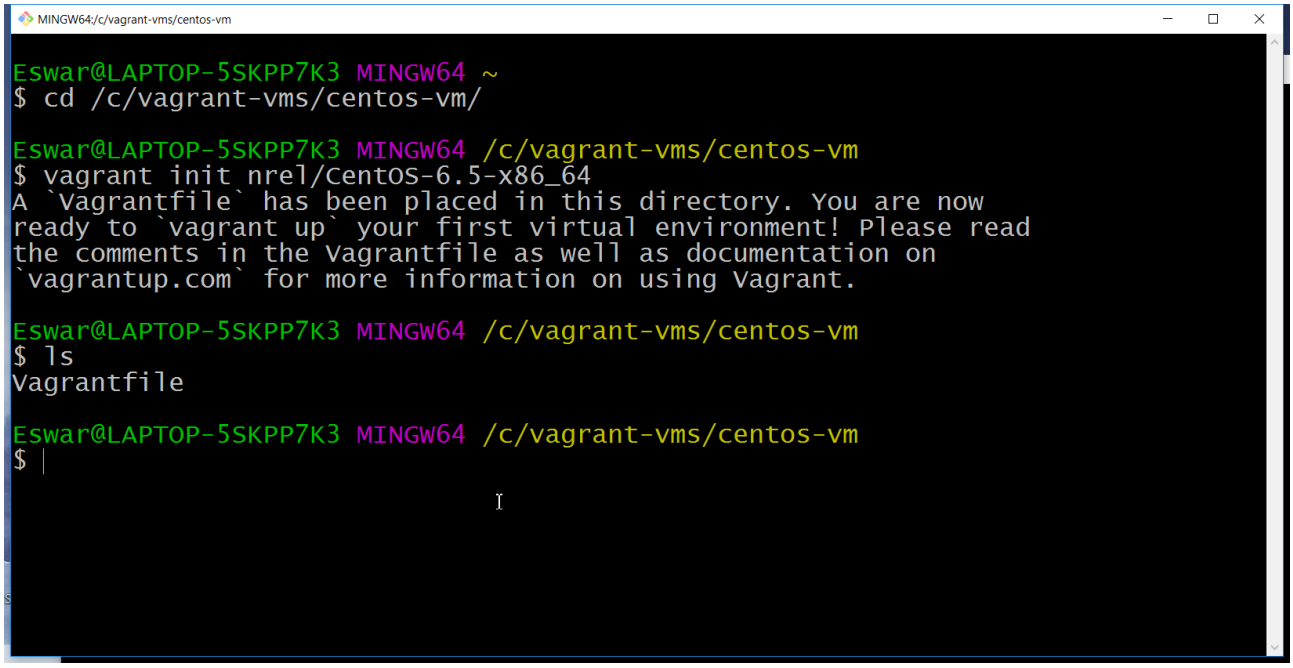


➔ **Open git bash.**

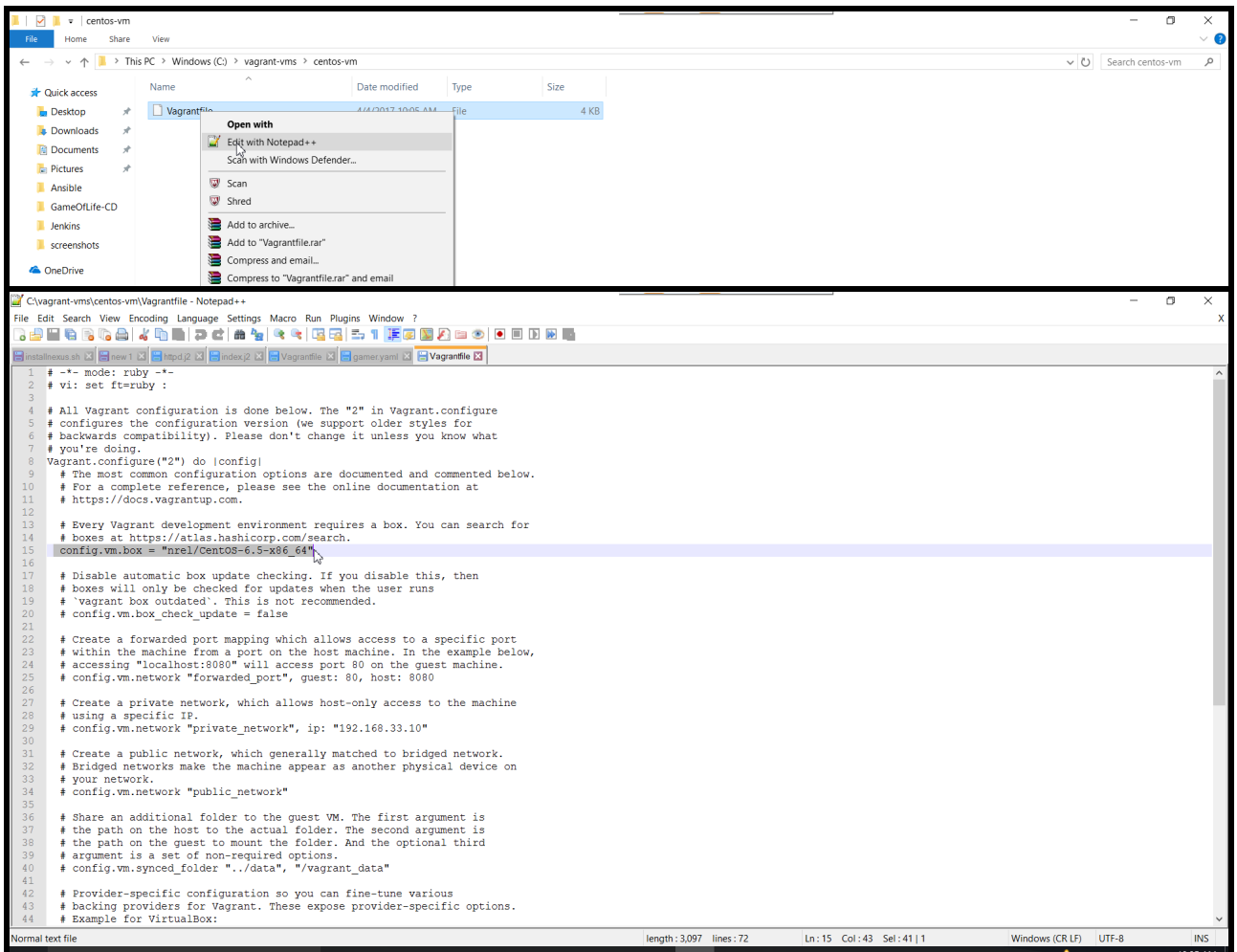


➔ **Follow the screenshot and place a vagrant file in centos-vm folder.**

1) cd /c/vagrant-vms/centos-vm
2) vagrant init Centos-6.5-x86_64
3) ls

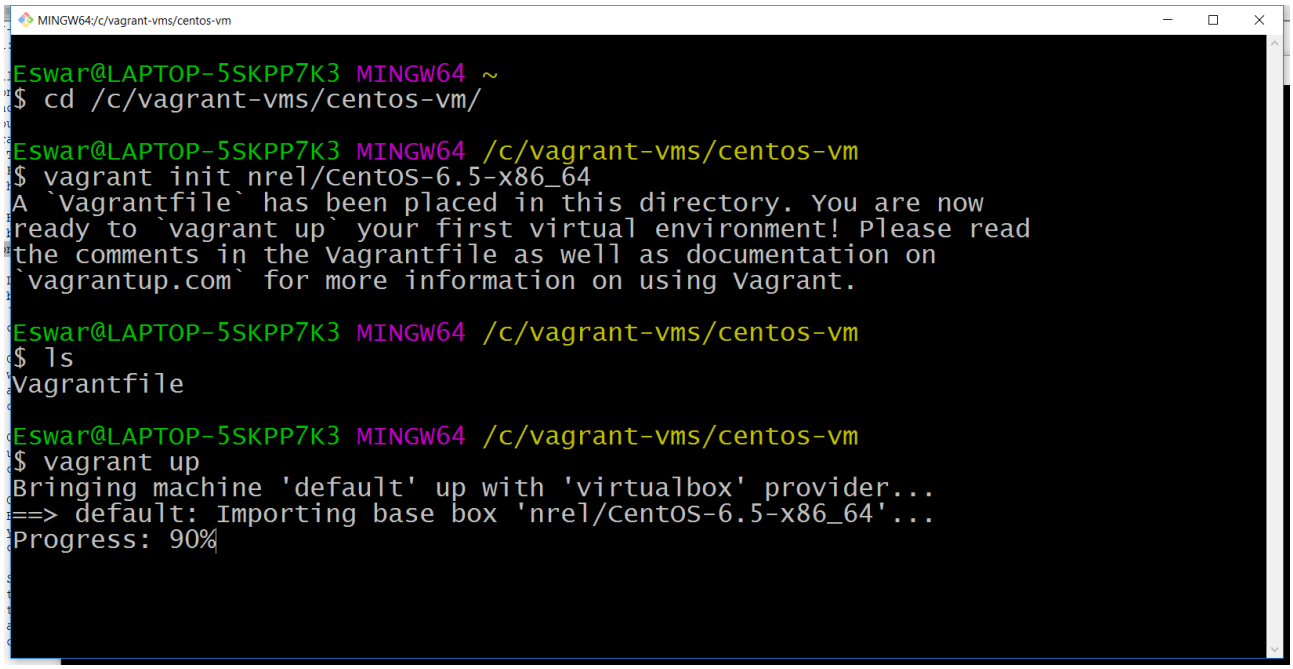


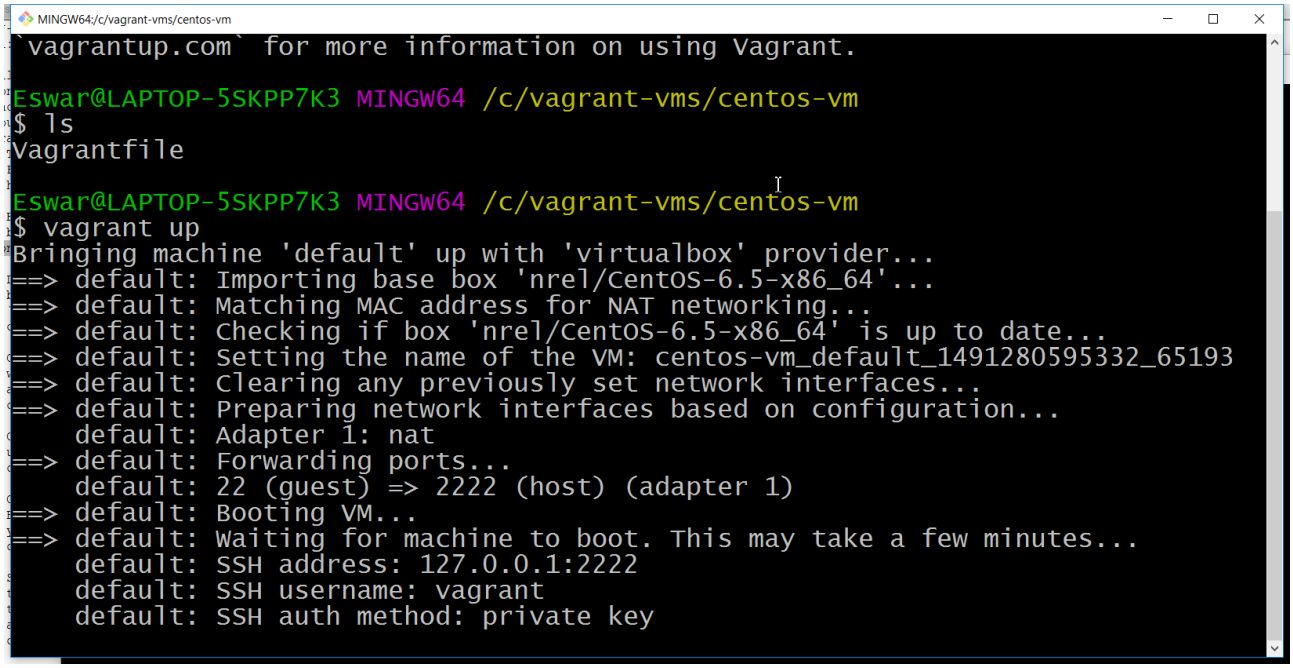
➔ **Go to centos-vm folder and check the file => Open it with Notepad++ editor.**



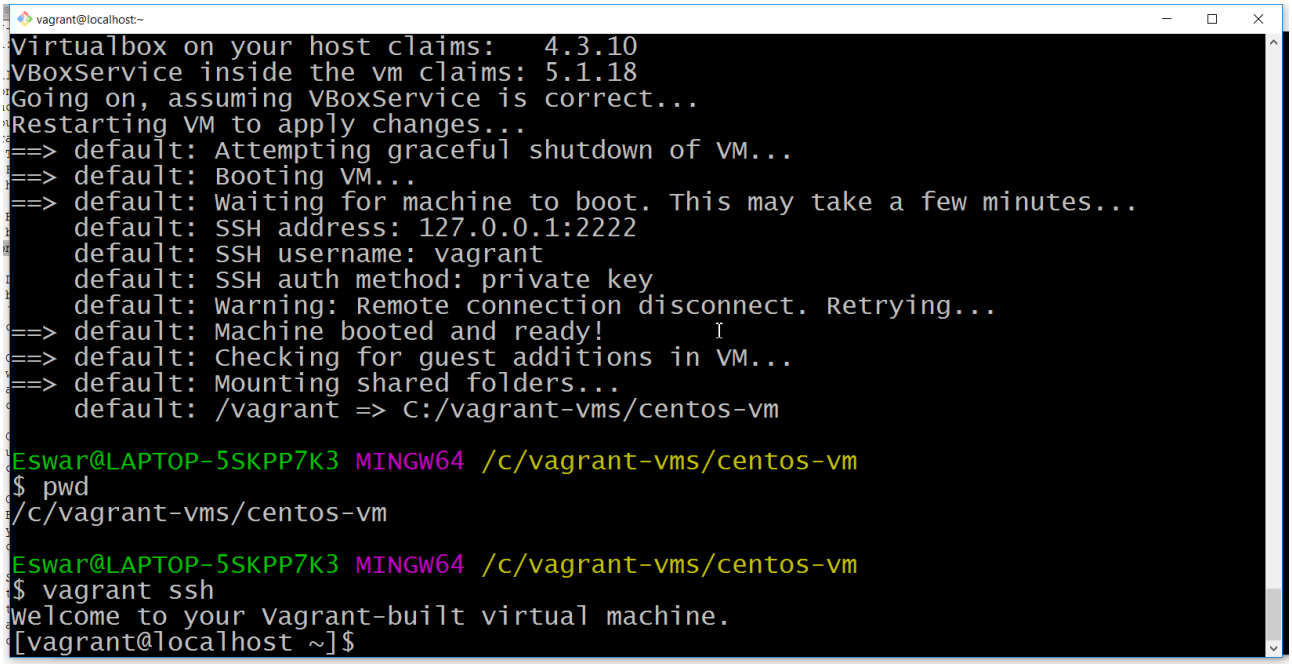
➔ **Close notepad++ after verification & Open git bash again.**

➔ **Go to centos-vm folder from git bash and run “vagrant up” command.**



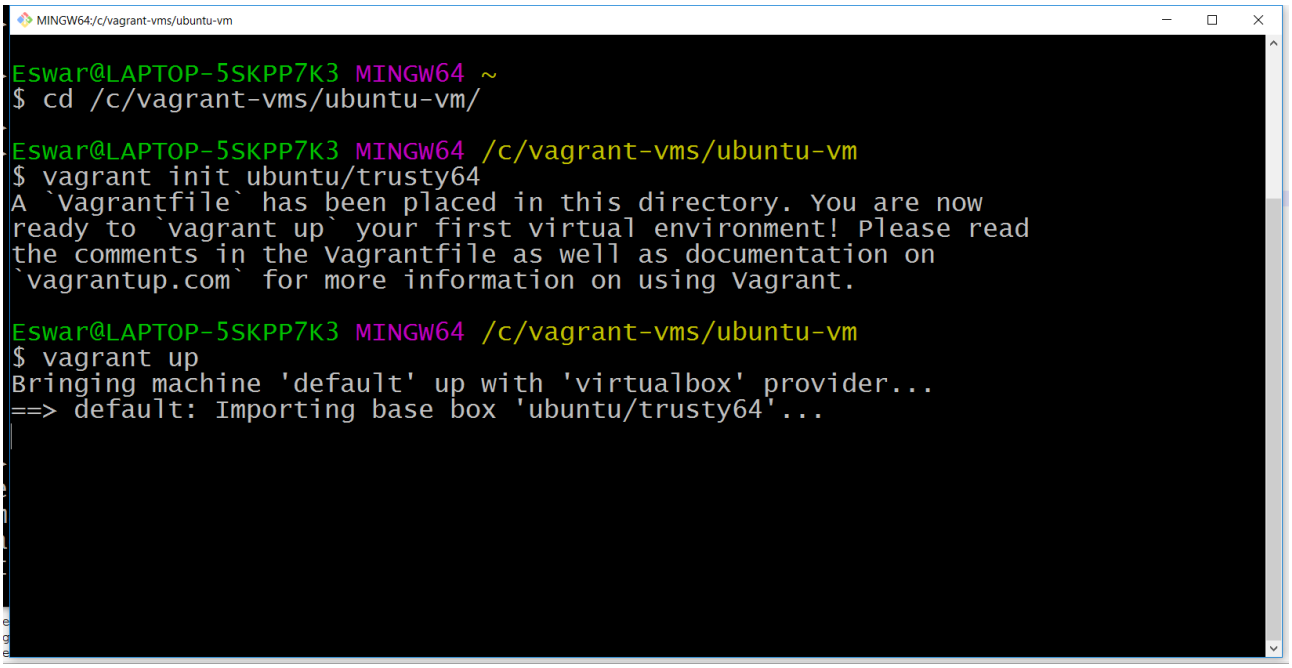


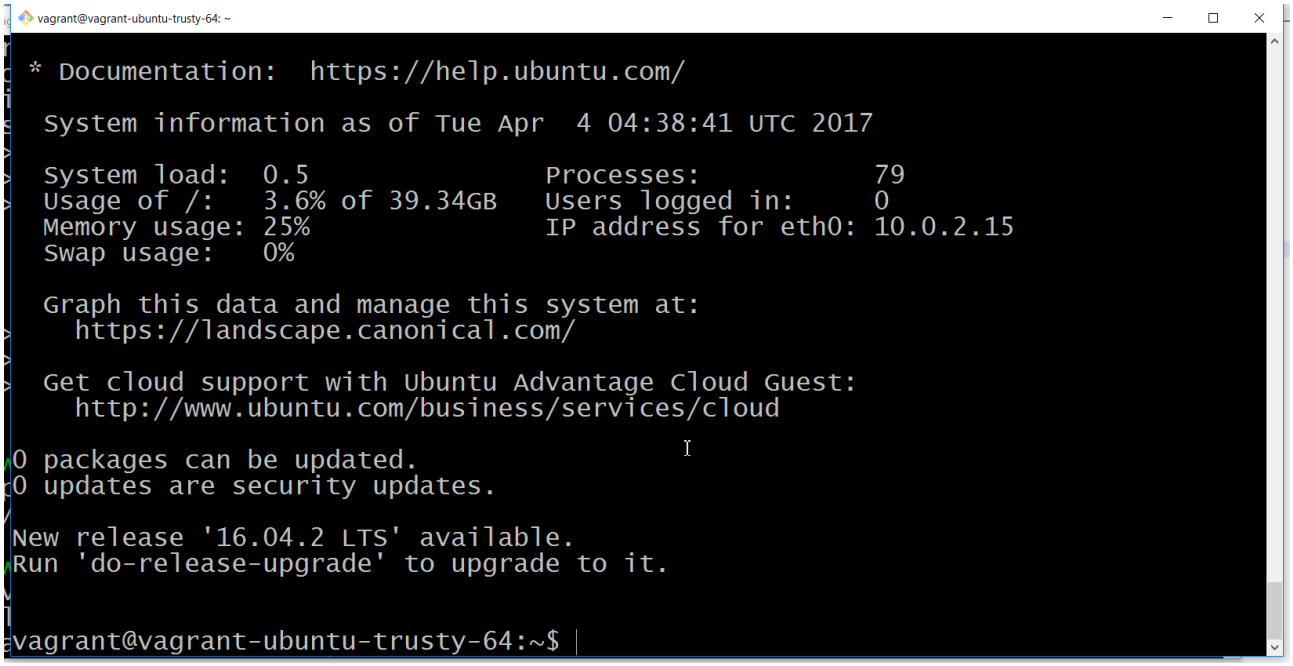
➔ **Once the vm is up login to the vm by giving “vagrant ssh” command.**



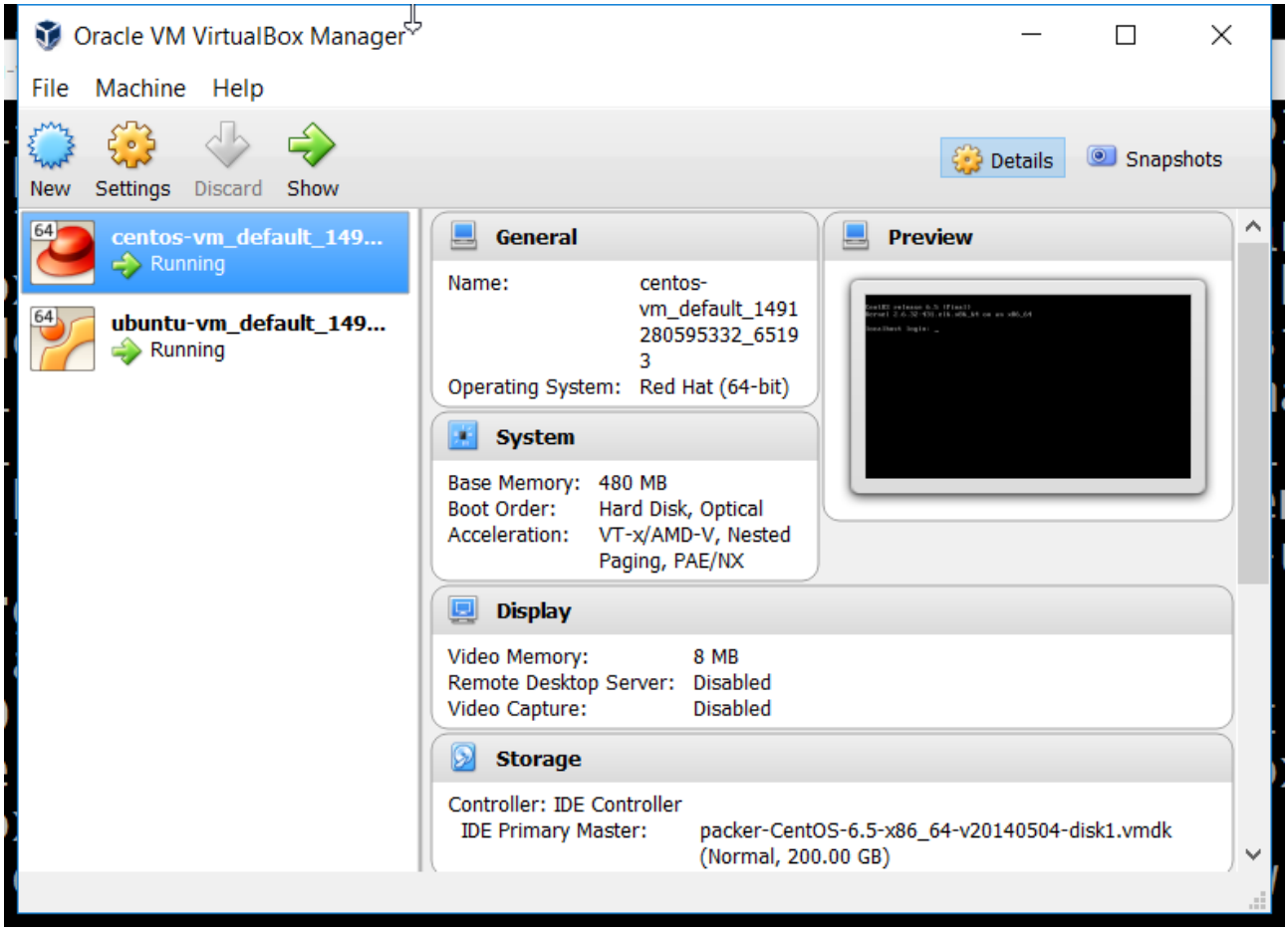
➔ **Follow the same procedure in ubuntu-vm folder, Open another git bash prompt.**

1) cd /c/vagrant-vms/ubuntu-vm
2) vagrant init ubuntu/trusty64
3) vagrant up



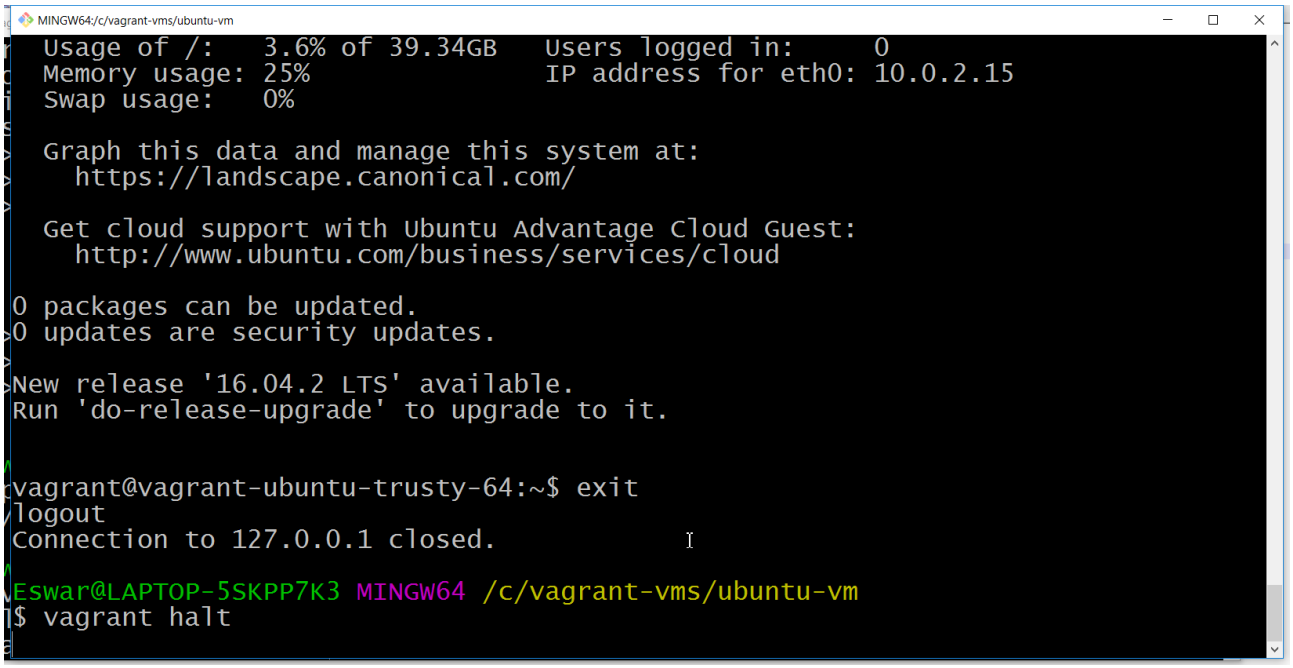


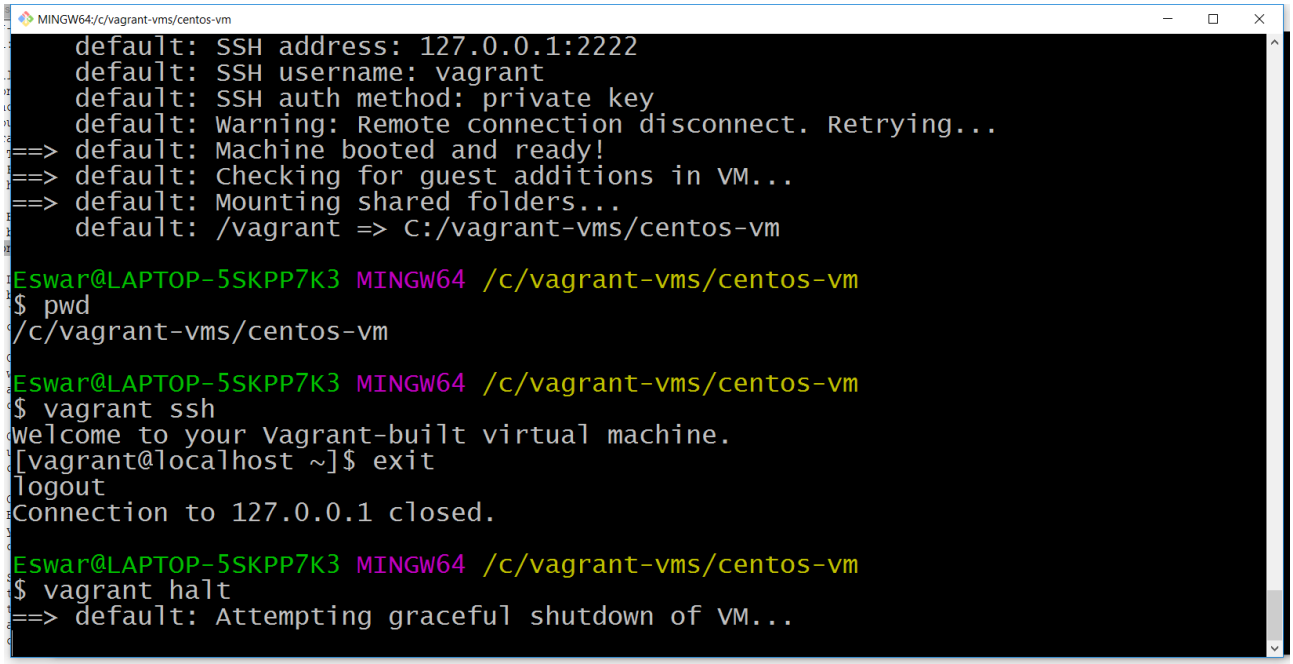
➔ **Open virtualbox software and you will see two vm's running there.**



➔ **Once you are done exploring the vm, you can shut it down.**

➔ **Hit “exit” command to come out of the vm shell => hit “vagrant halt” to shut it down.**





➔ **Check in the virtualbox, both the vm's will be powered off.**