

## **Microsoft Deployment Toolkit (MDT)**

The purpose of the MDT is to help automate the deployment of Windows operating systems and applications to computers in the environment. At high level, MDT automates the deployment process by configuring the unattended Setup files for Windows and packaging the necessary files into a consolidated image file that you then deploy to reference and target computers.

### **Reasons for using MDT**

Free software tool

Automation creation of golden images with scripting

### **Prerequisite**

Domain Controller

DNS

Installing MDT on deployment computer

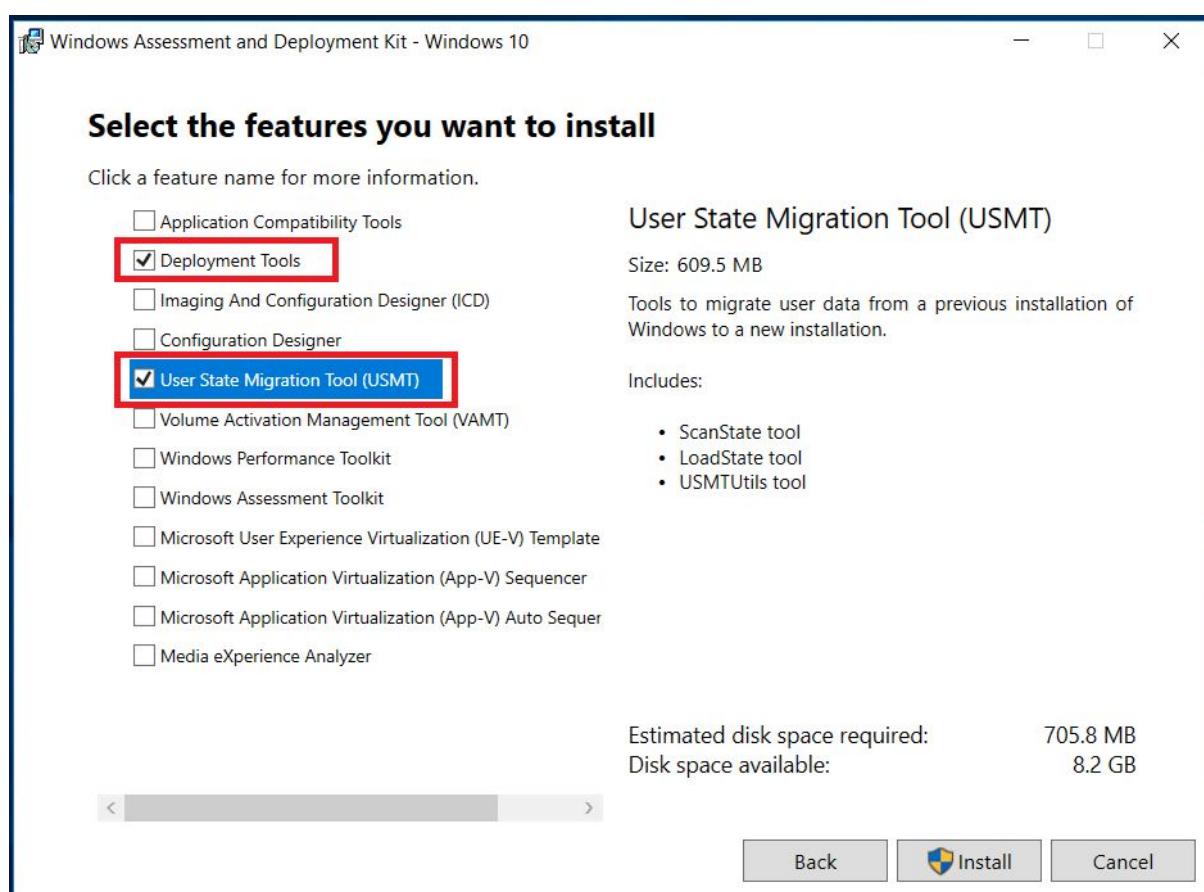
Installing Windows Assessment and Deployment Kit (Windows ADK) on deployment computer

### **Install Windows ADK - 3 Features needed:**

#### Main Installer:

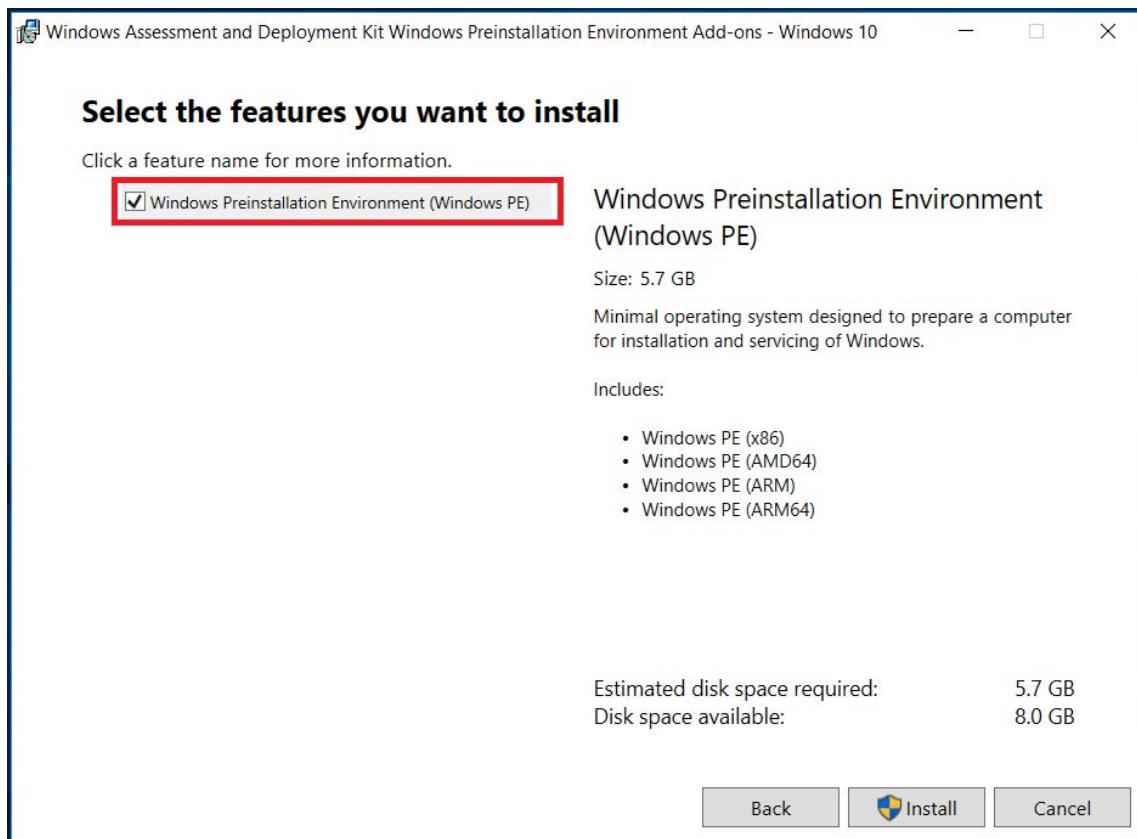
Deployment Tools

User State Migration Tool



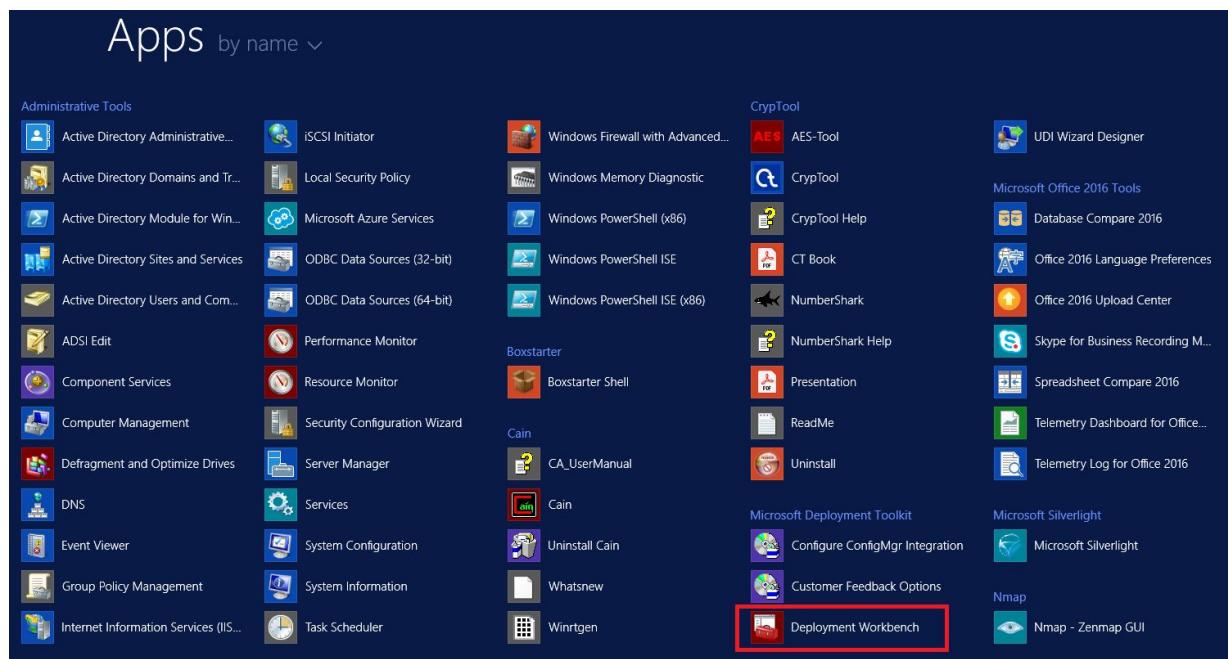
Add-ons installer:

Window PE

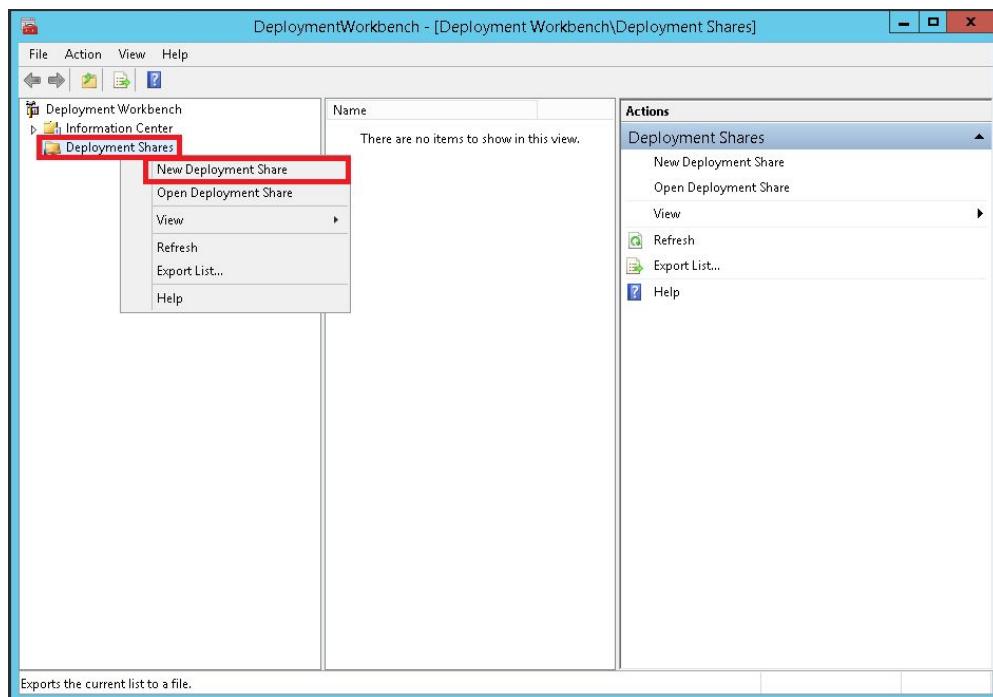


**Setting up MDT**

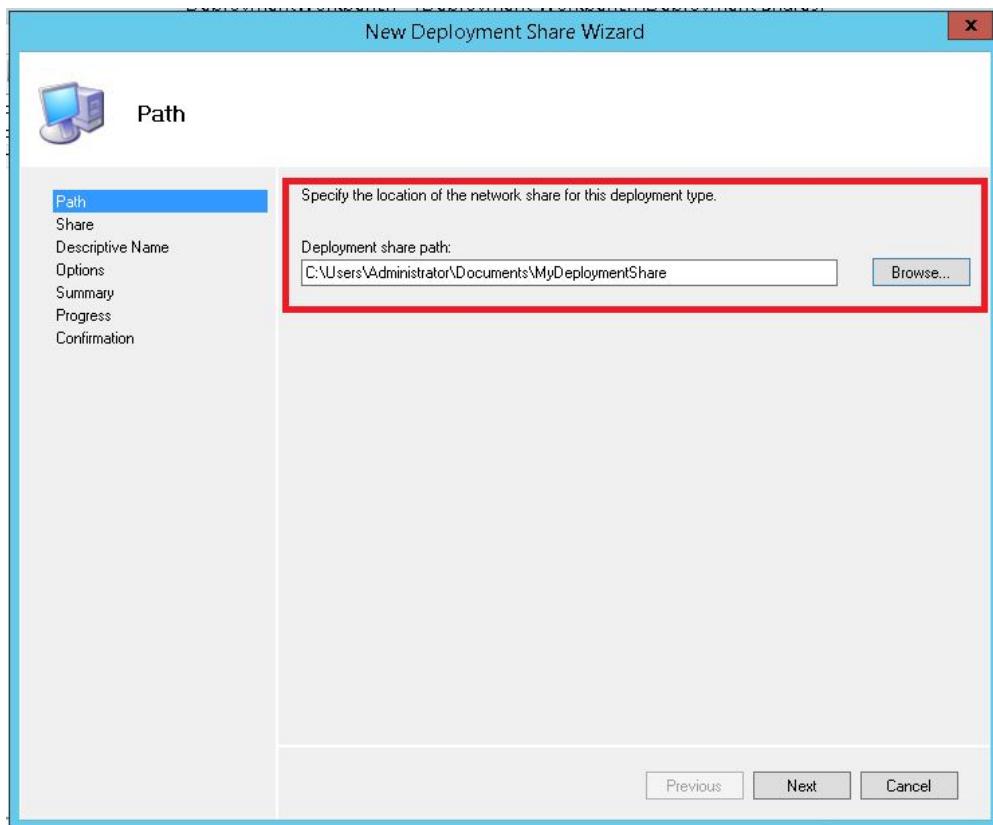
After installing MDT, open Deployment Workbench.



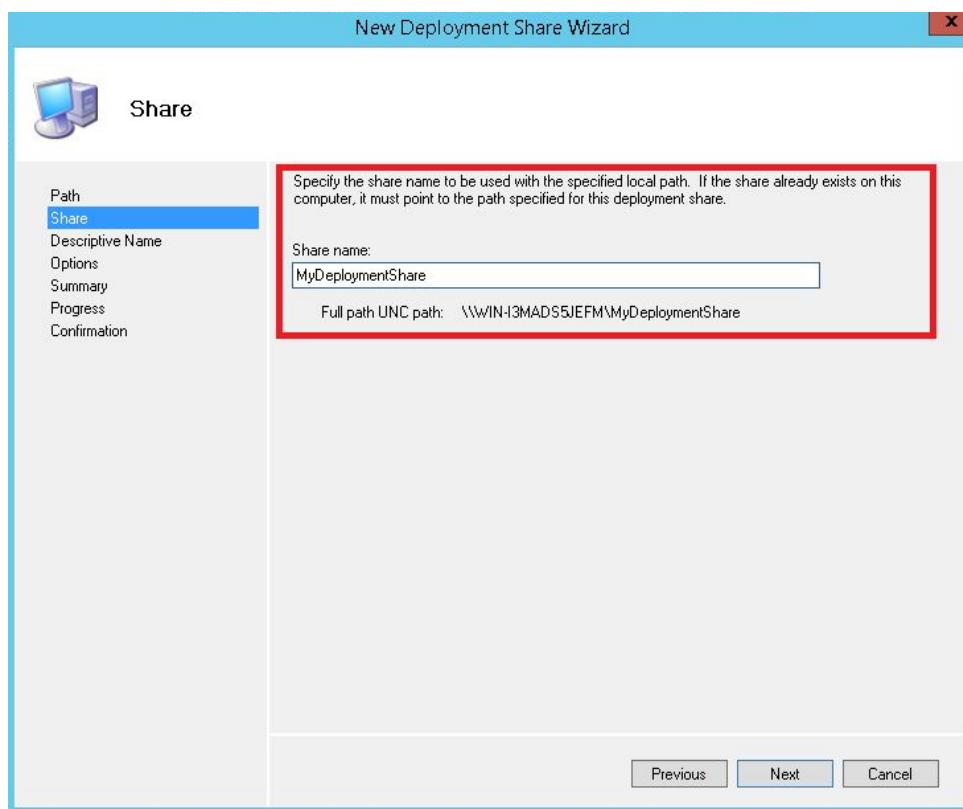
Create a new deployment share by right click on “Deployment Shares” and click on “New Deployment Share”.



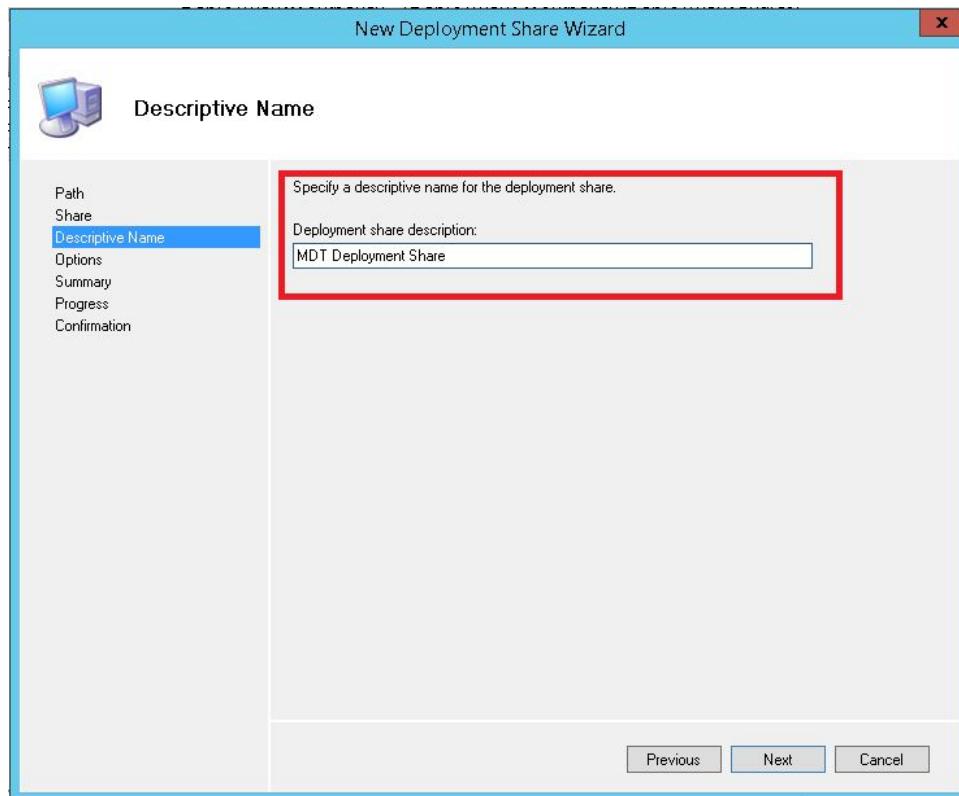
Specify the location of the folder for the deployment share and click next.



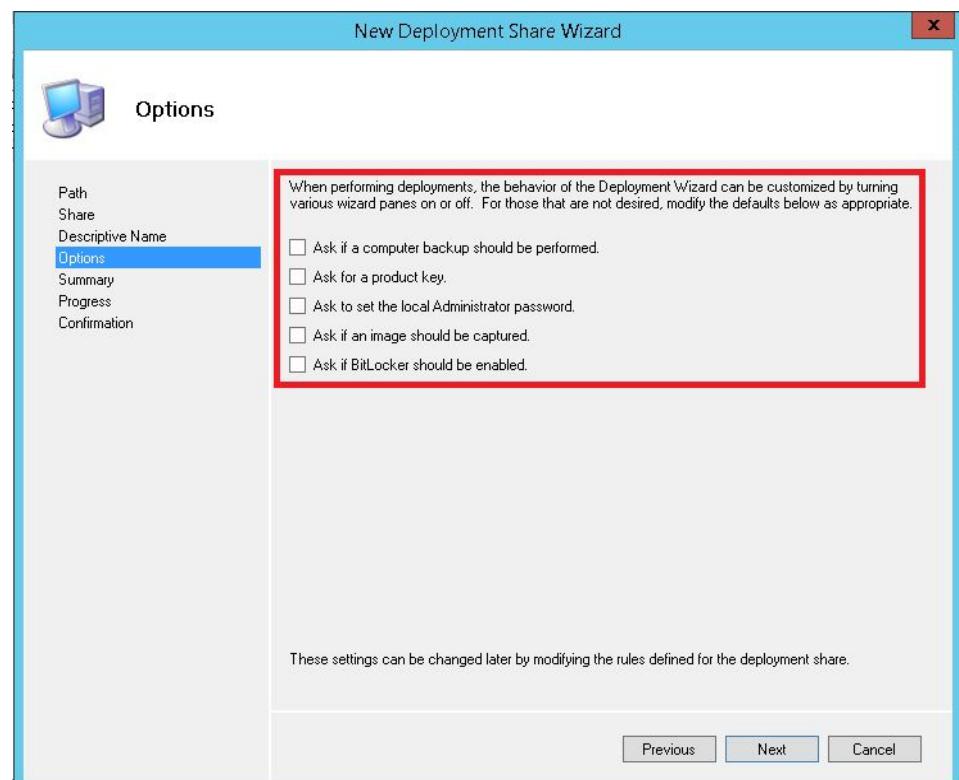
Specify the share name and the UNC path will be given for network share. Click next.



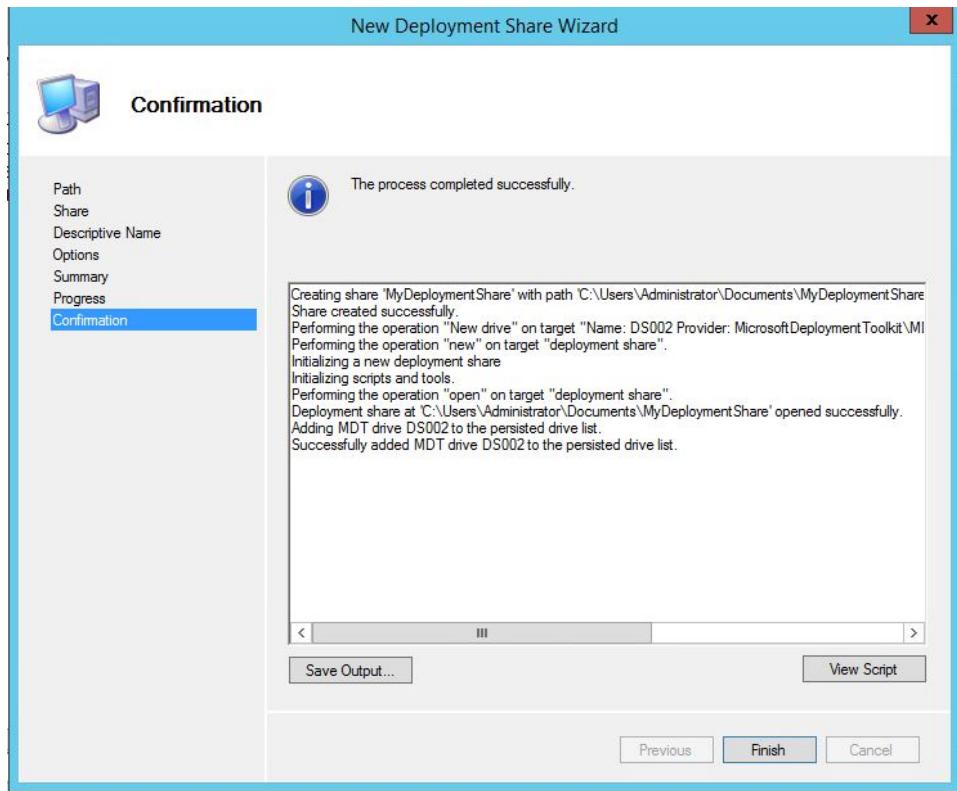
Specify a name for the folder on MDT and click next.



Uncheck all options and click next.

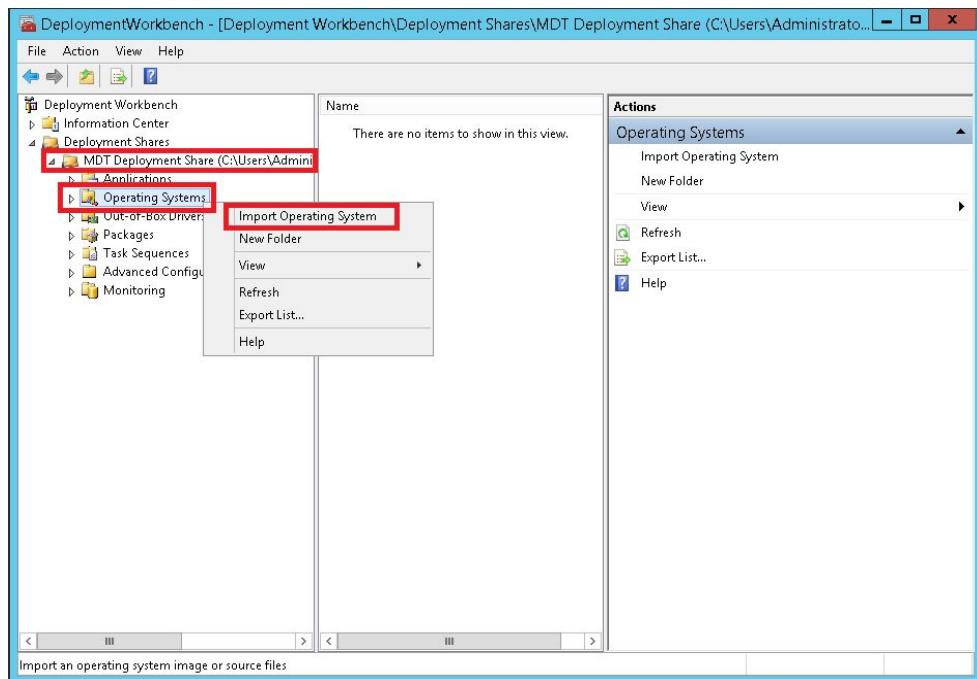


Click next on the summary page, and wait for the folder to be ready. Click finish.

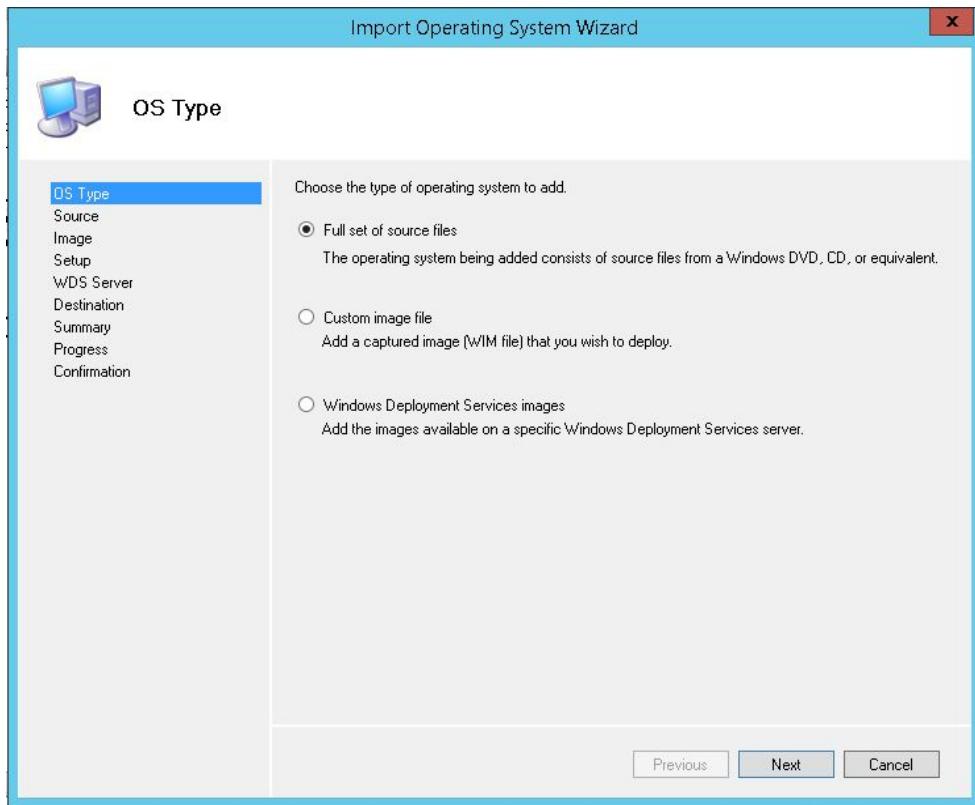


## Importing OS into MDT

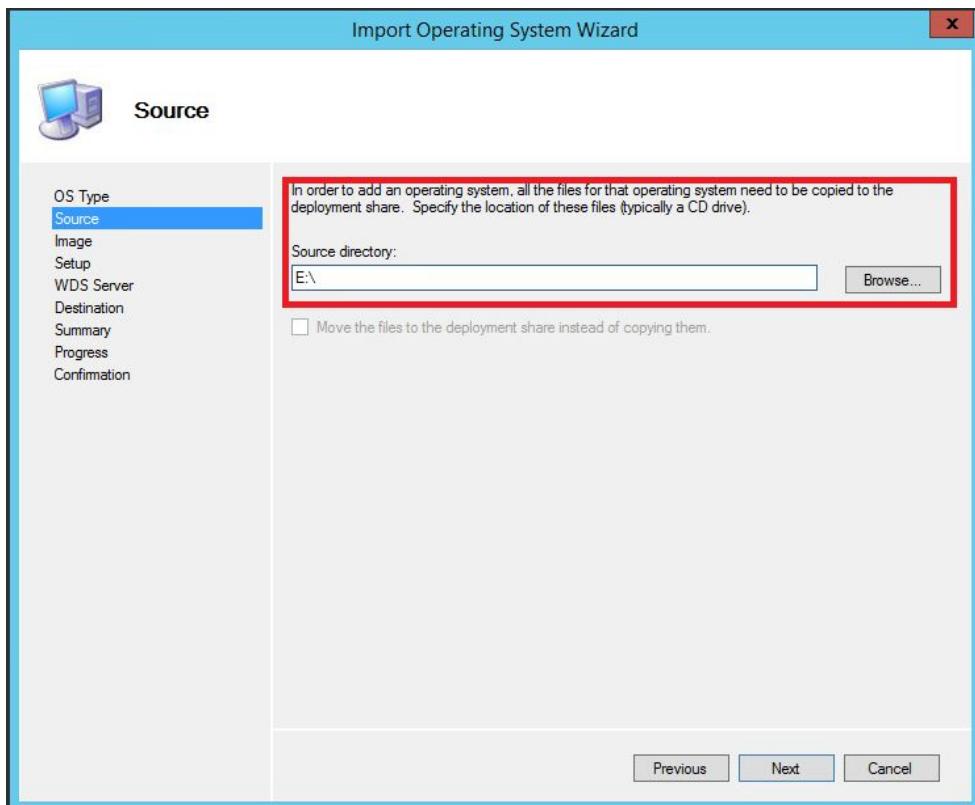
Expand the folder you have named on the MDT. Right click on “Operating Systems” and click on “Import Operating System”.



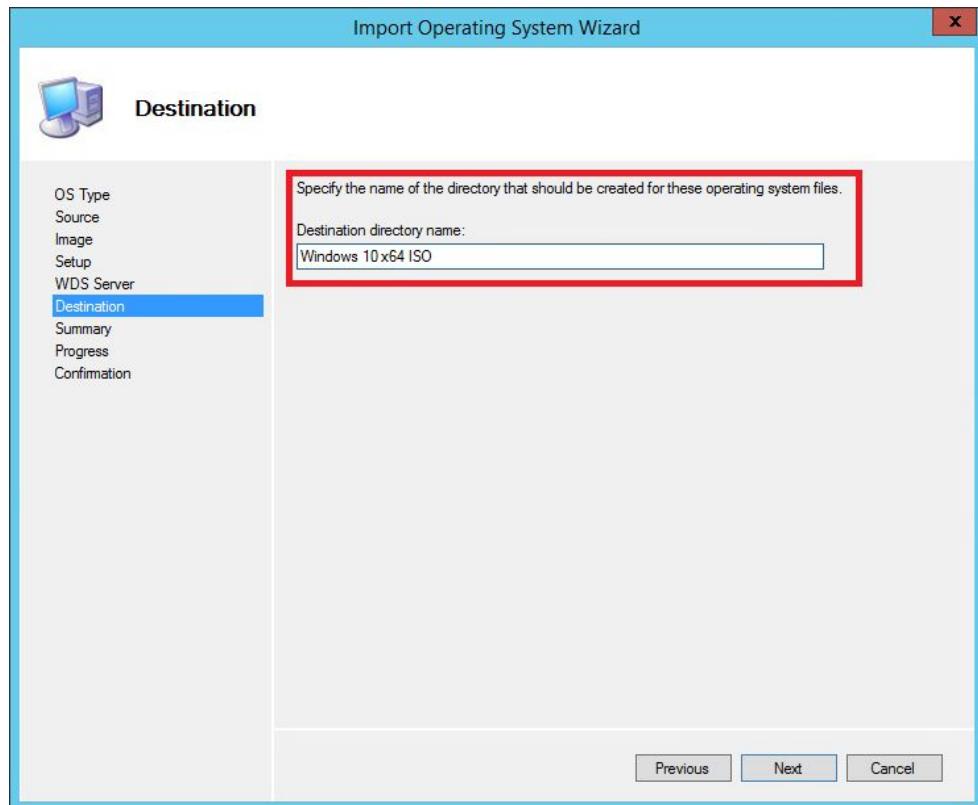
Select the type of image you want to add and click next. In this example, I am using ISO images (Windows 7 & 10).



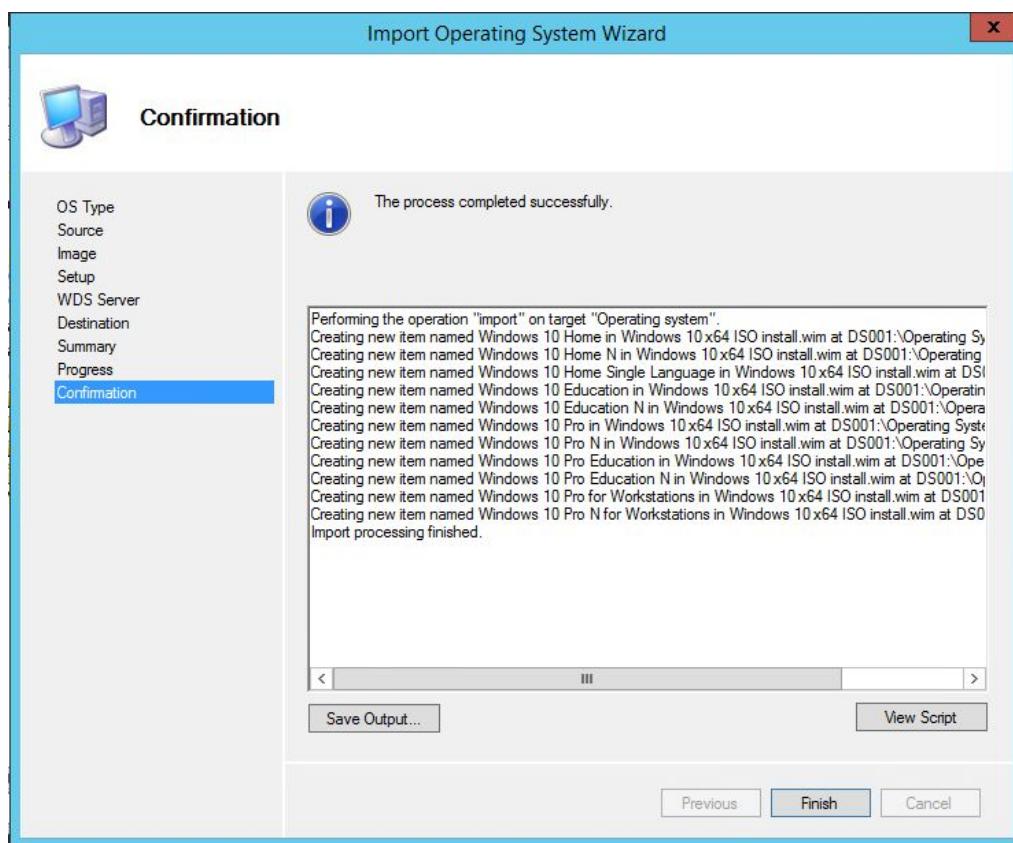
Locate the image source path and click next. In this example, I had mounted the ISO image.



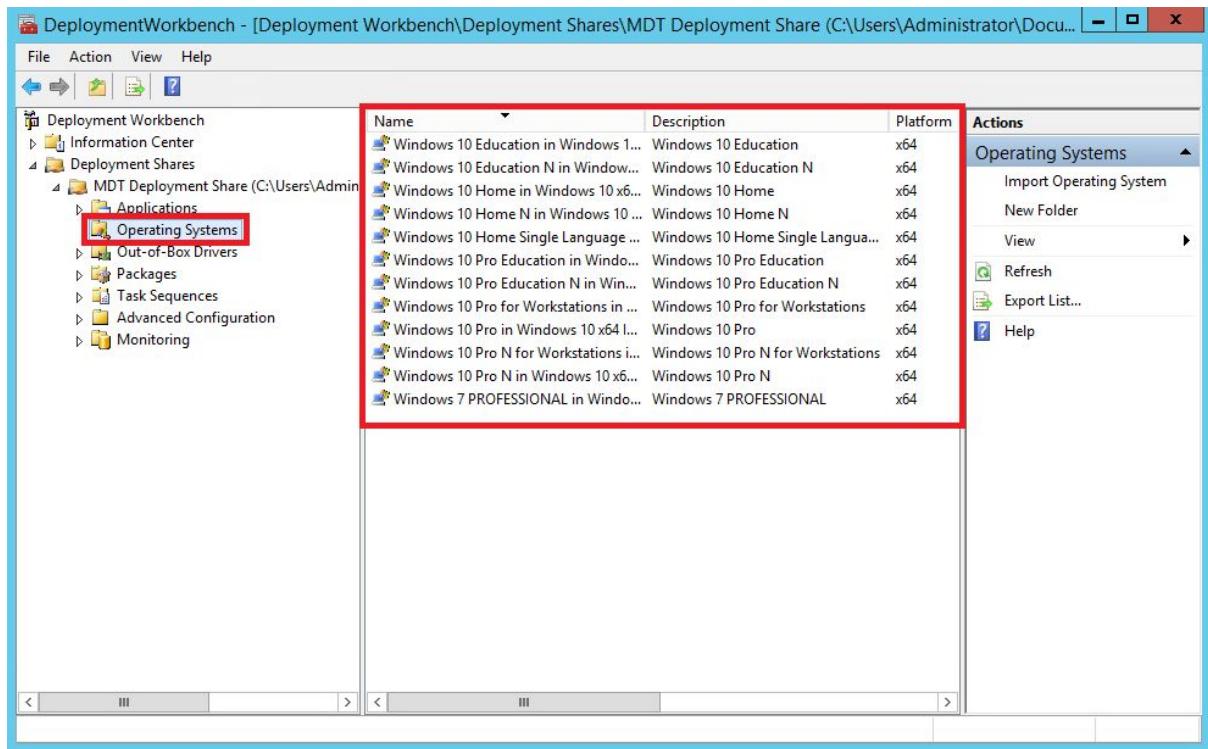
Specify the name of the directory to store the imported OS and click next.



Click next on summary page and wait for the OS to be imported to MDT. Click finish.

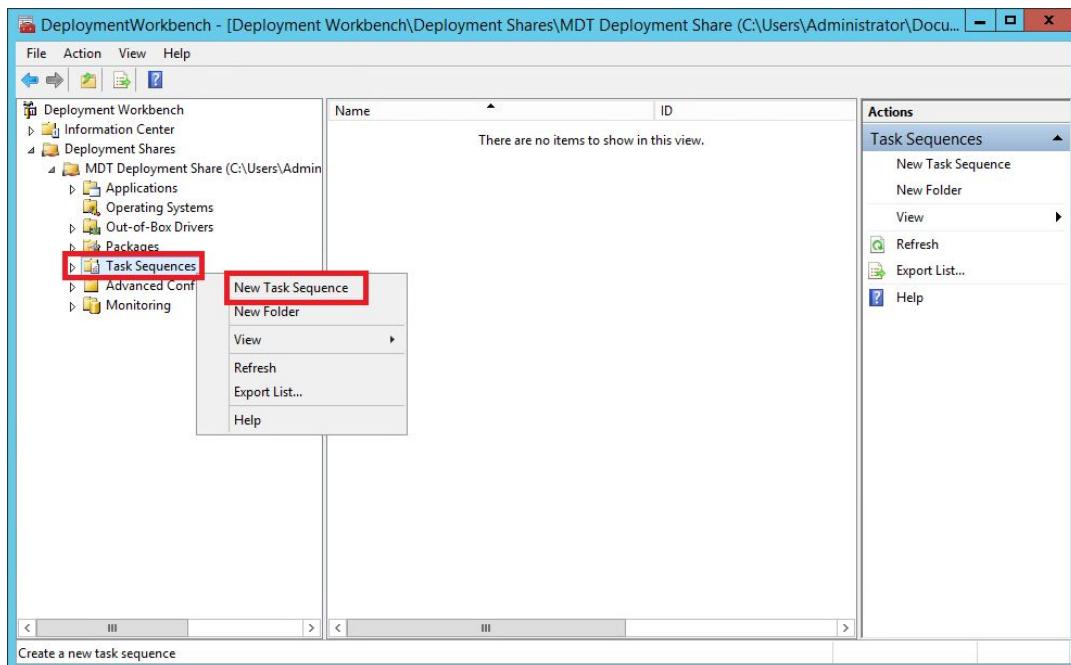


You can check your imported ISO images under “Operating Systems”.

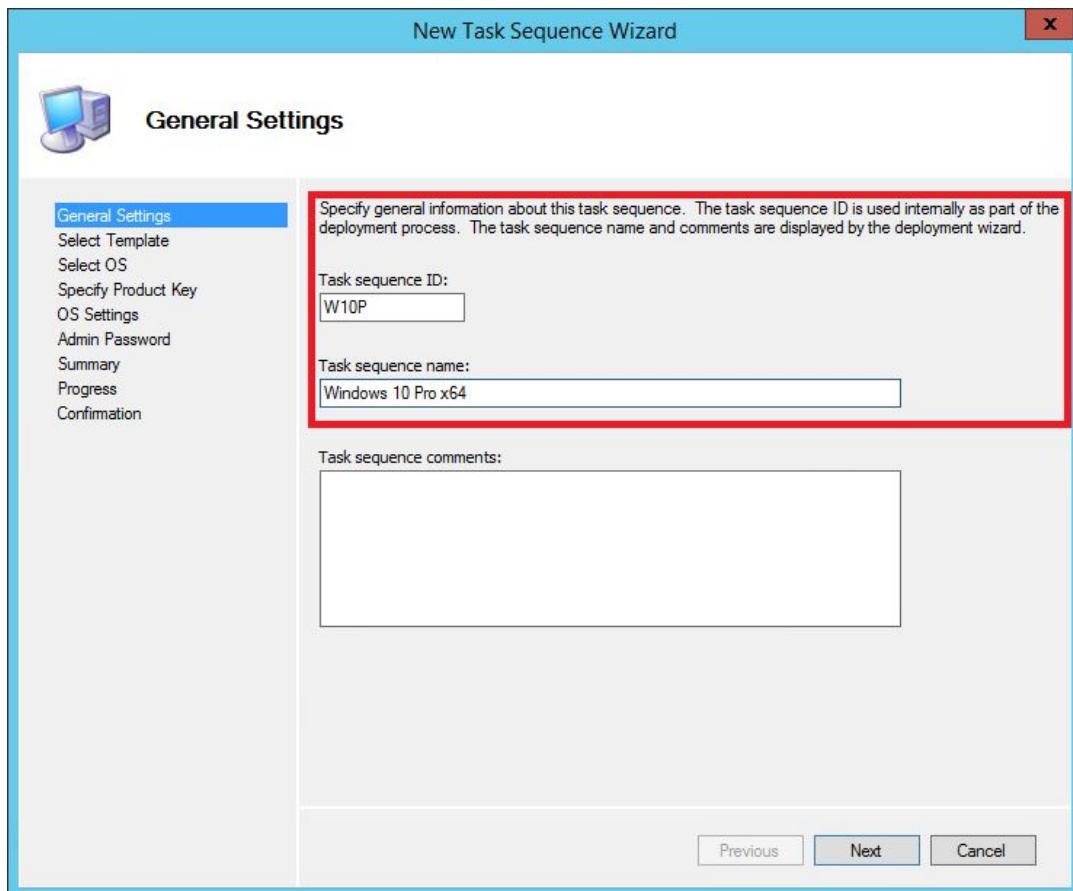


## Creating Task Sequences

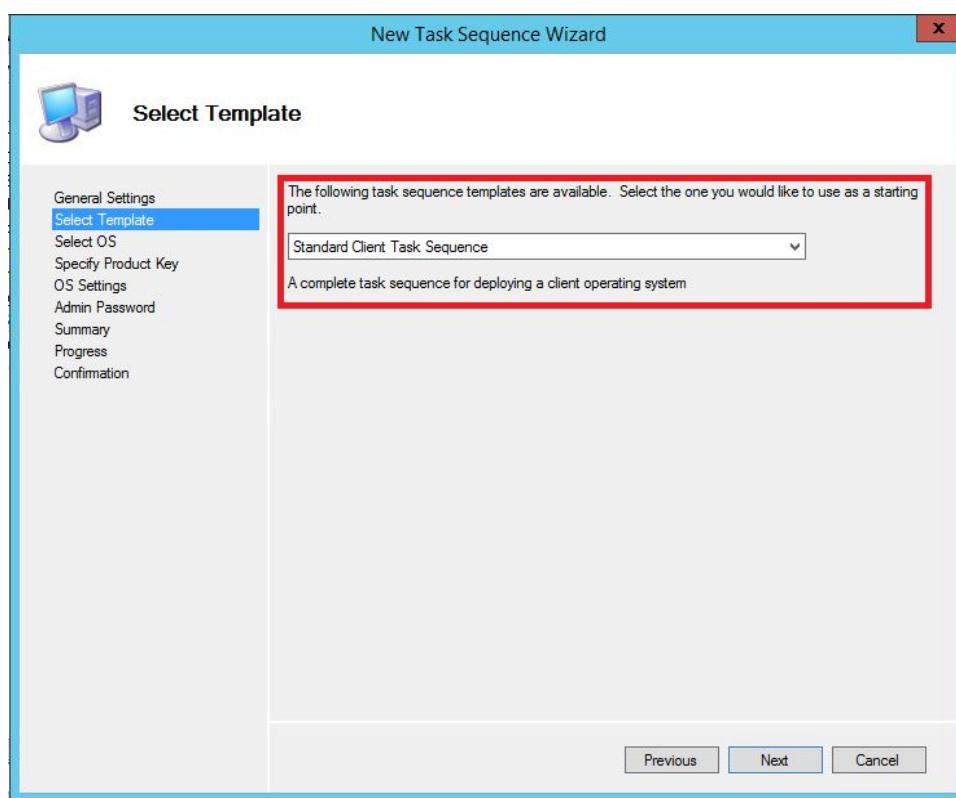
Right click on “Task Sequences” and click on “New Task Sequence”.



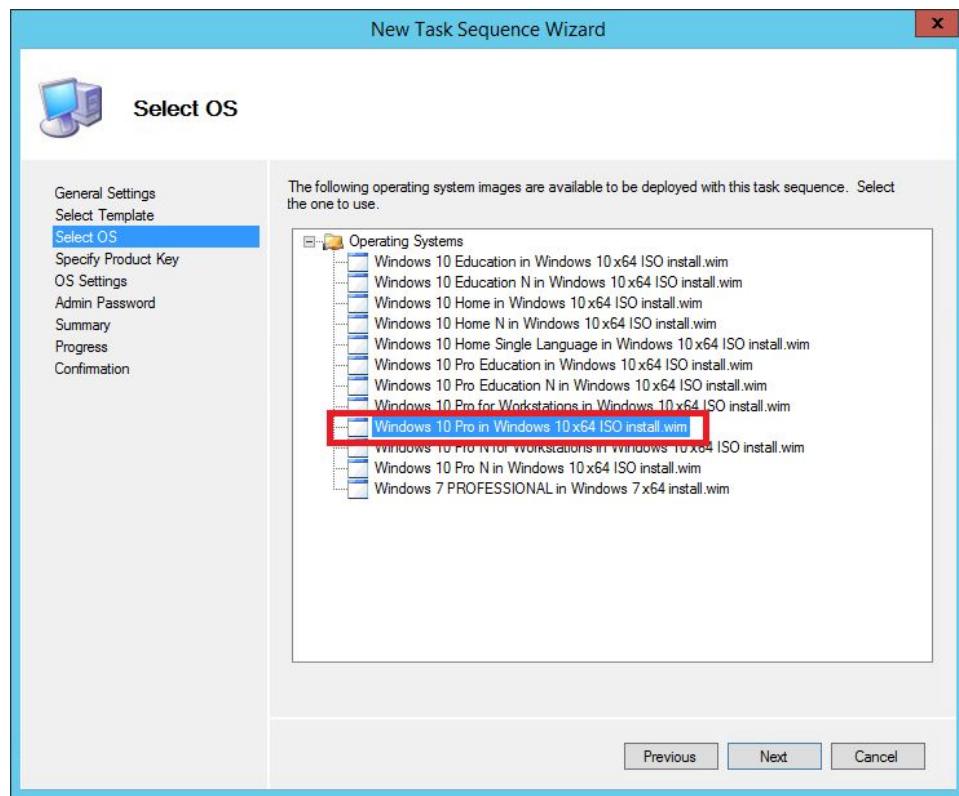
Specify the Task sequence ID and Task sequence name and click next.



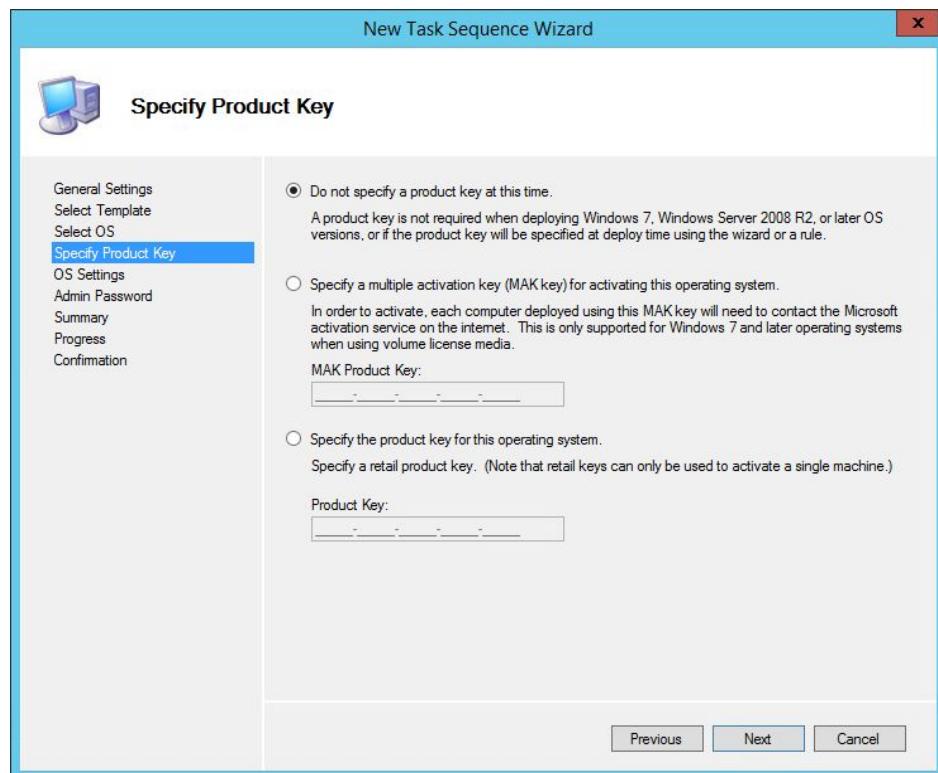
Select the templates that you want to use and click next. In this example, I selected Standard Client Task Sequence to deploy OS.



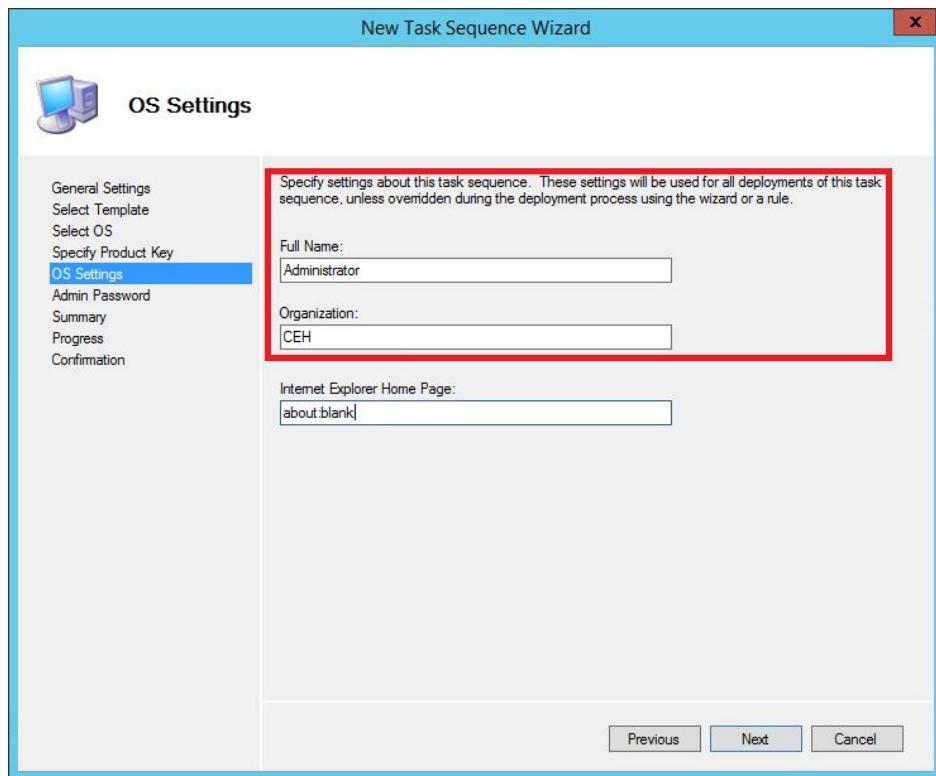
Select the OS to deploy and click next. In this example, I selected Windows 10 Pro.



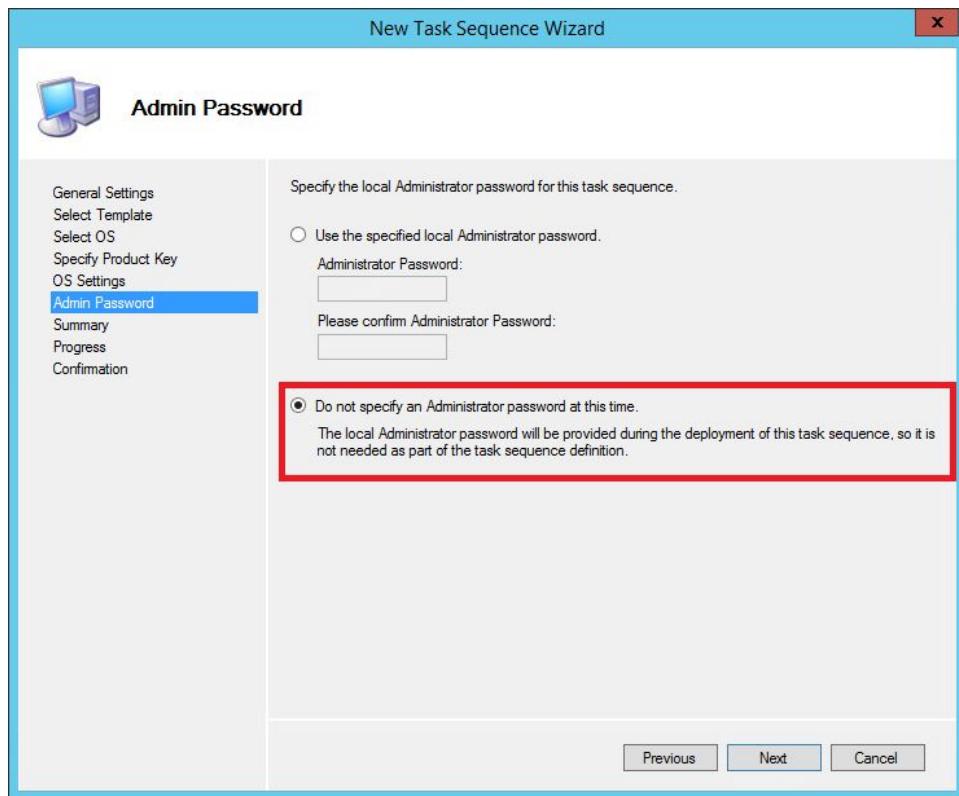
Select the option you want and click next. In this example, I selected “Do not specify a product key at this time.”.



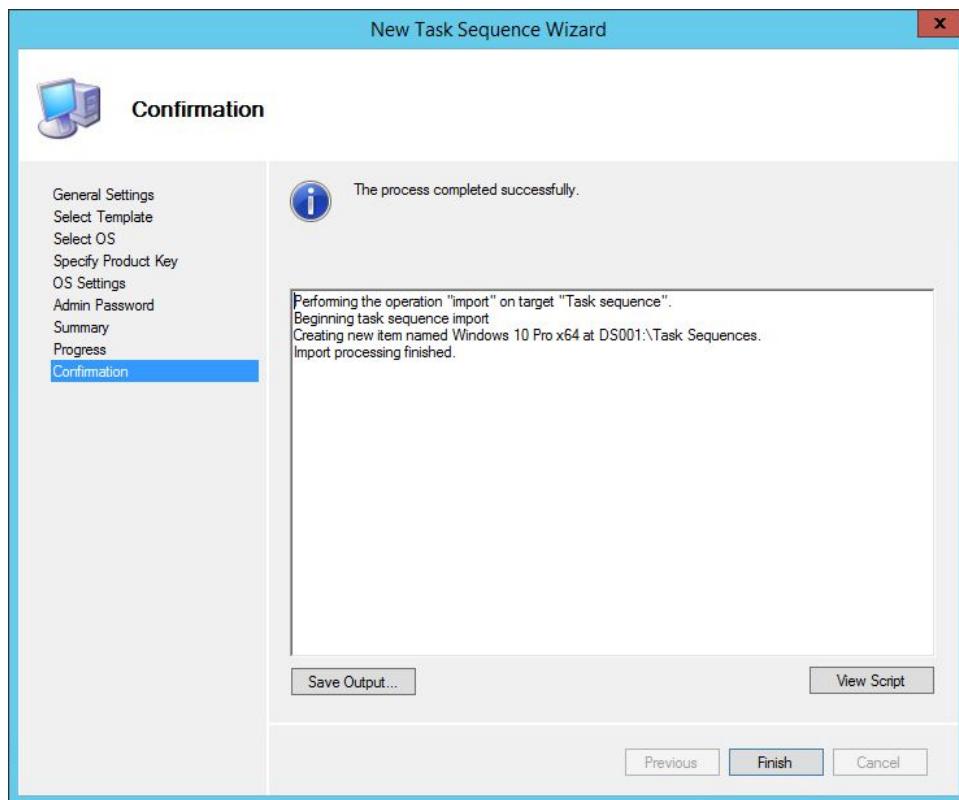
Specify the details about this task sequence you want (this doesn't affect the deployment) and click next.



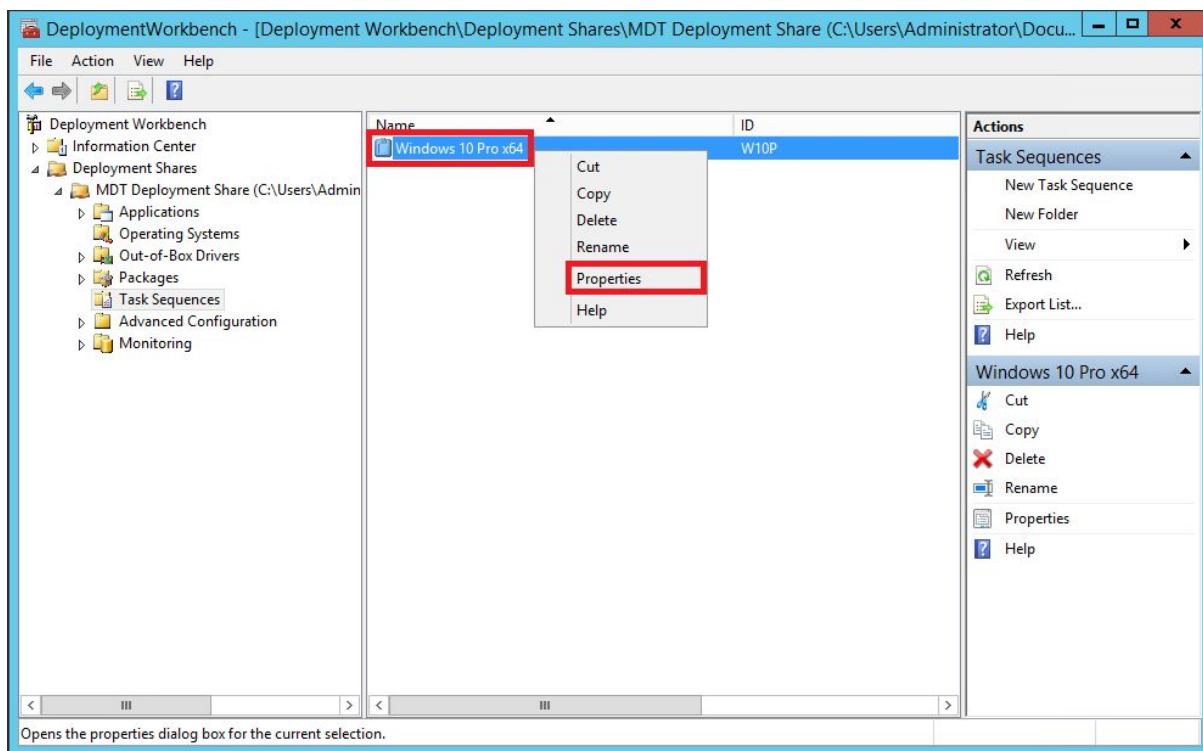
Select “Do not specify an Administrator password at this time.” as the password for the local administrator can be configured in the task.



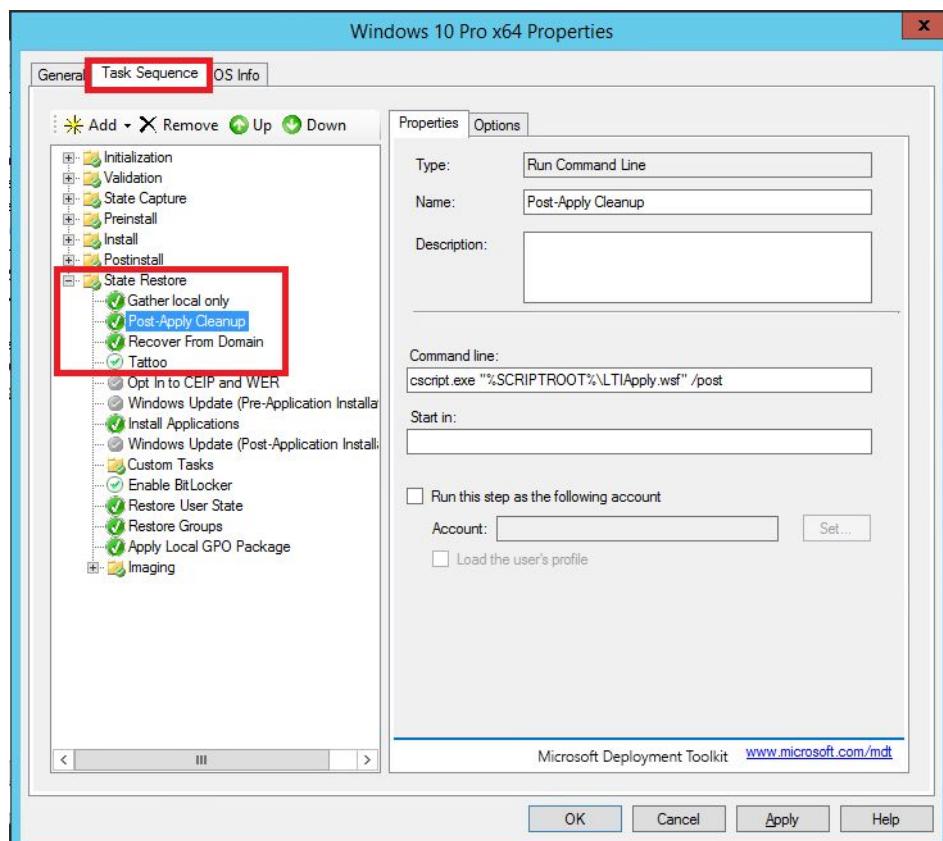
Click next on the summary page and wait for the task sequence to be created. Click finish.



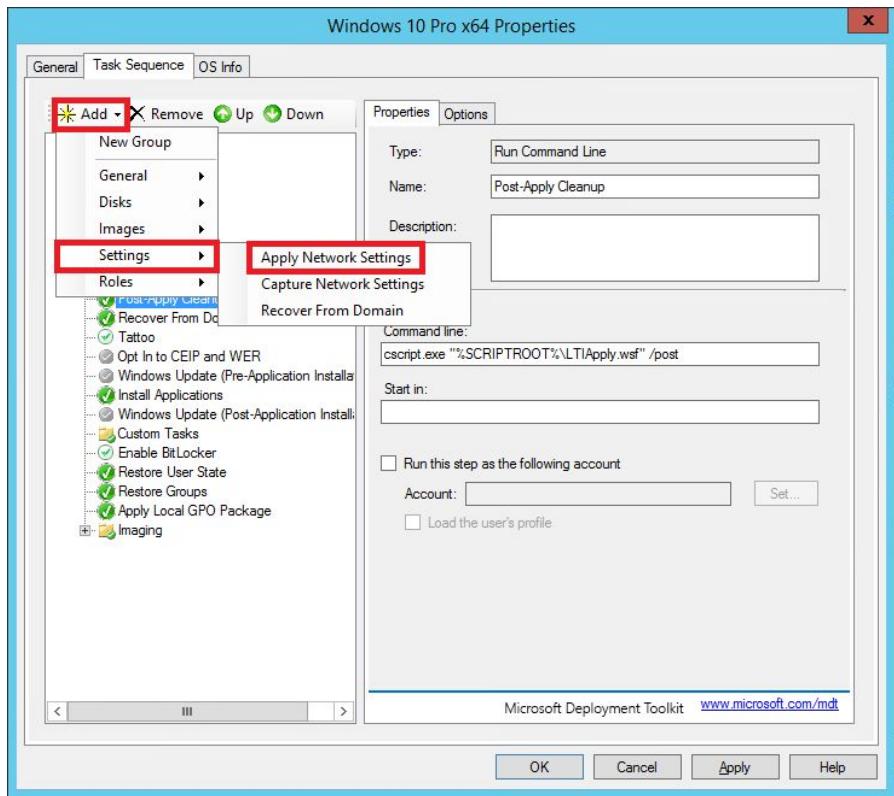
Right click on the task sequence you just created and click on “Properties”.



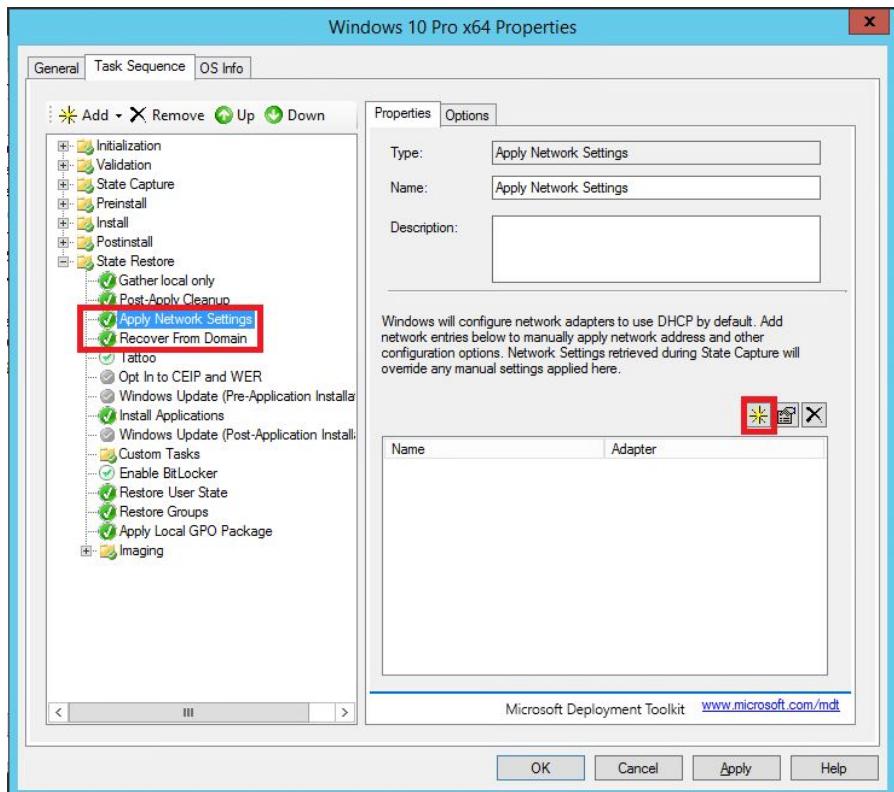
Click on “Task Sequence” tab, and find the task “Recover From Domain” under “State Restore”.



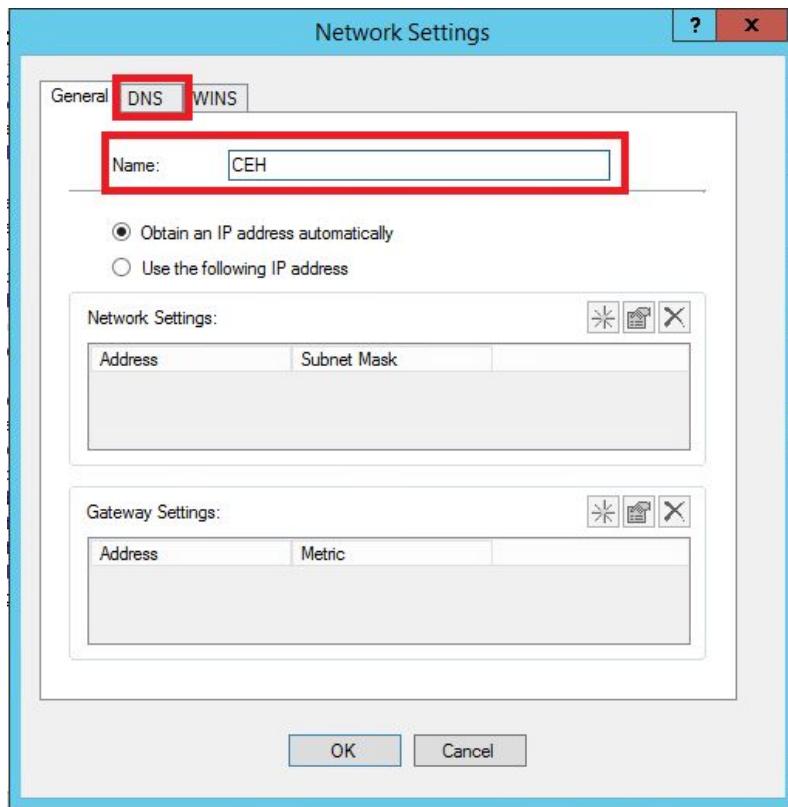
Add a task to include your DNS Server IP Address above “Recover From Domain” by clicking on “Add”, “Settings” and then “Apply Network Settings”.



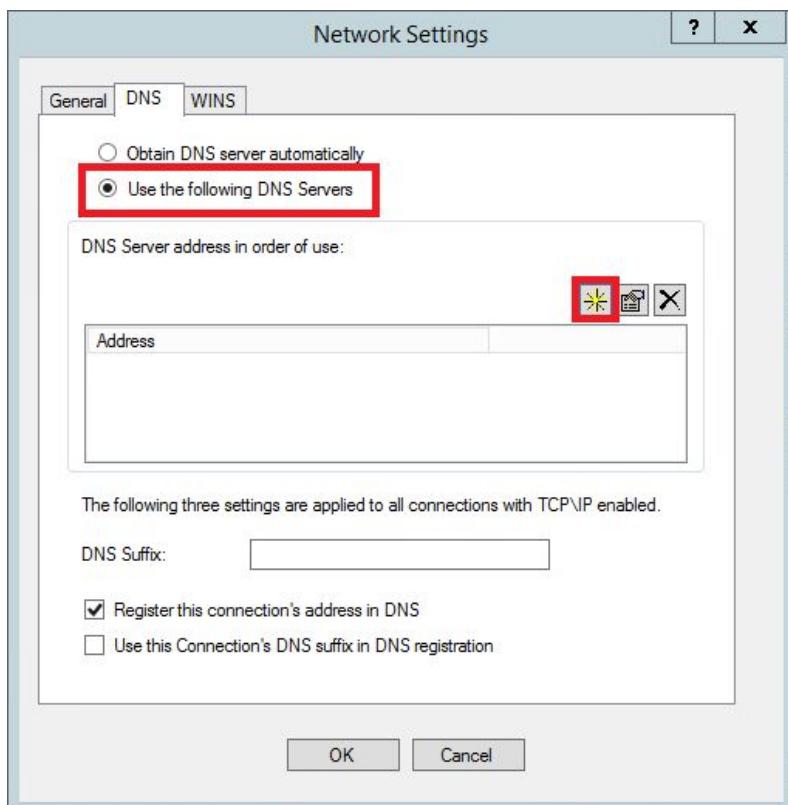
Click on the task you created and then click the add button.



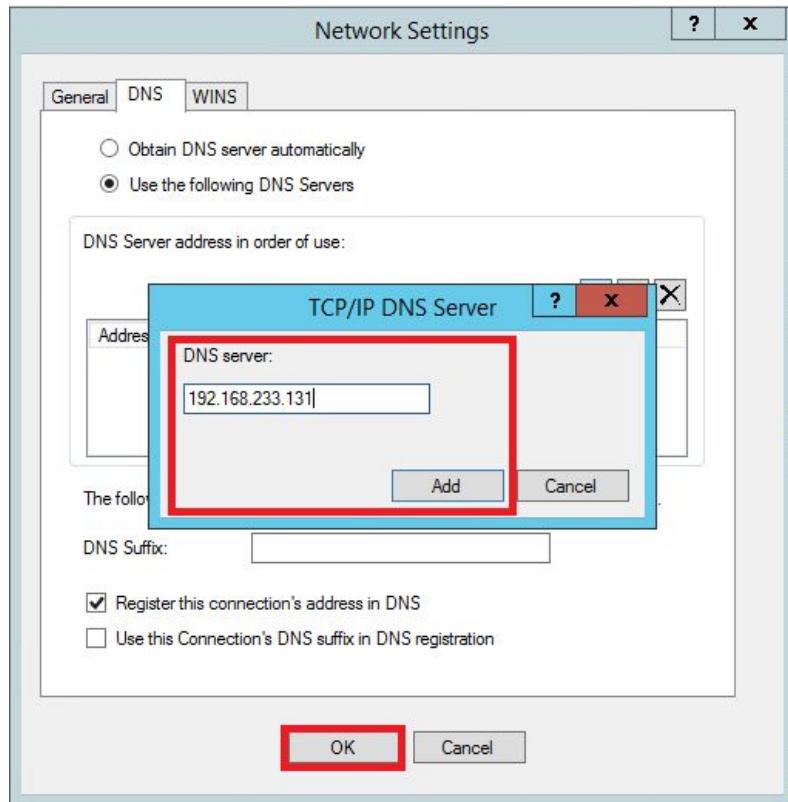
Specify your network name and click on “DNS” tab.



Select “Use the following DNS Servers” and click the add button.

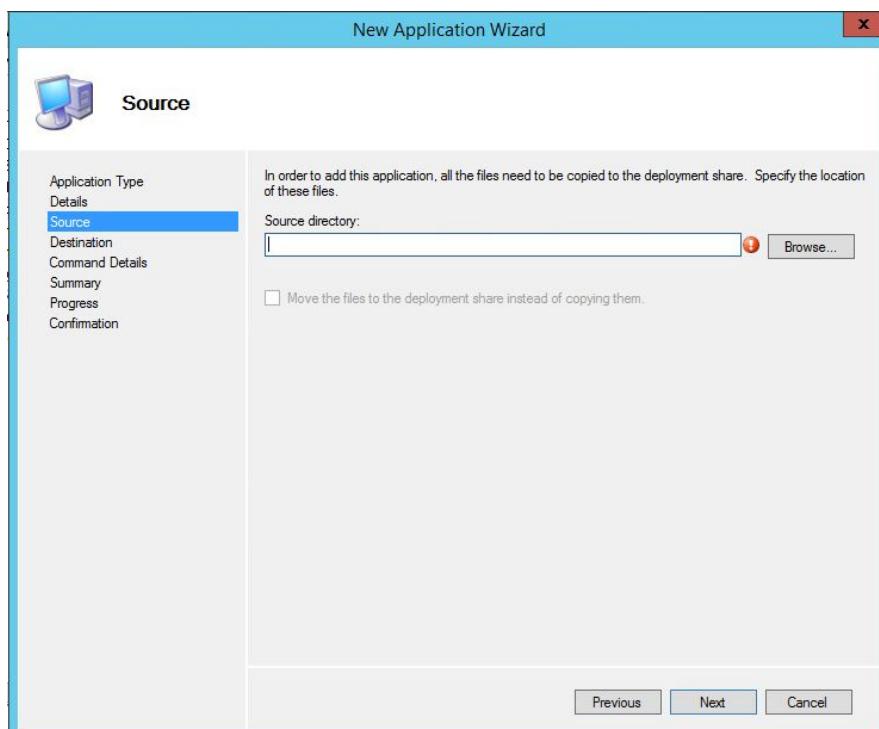


Specify your DNS IP Address and click ok.



## Including Applications Installation

There are 2 ways to including applications installation into the deployment. One of the ways is to use the MDT's application scripts provided which requires you to add the setup .exe.

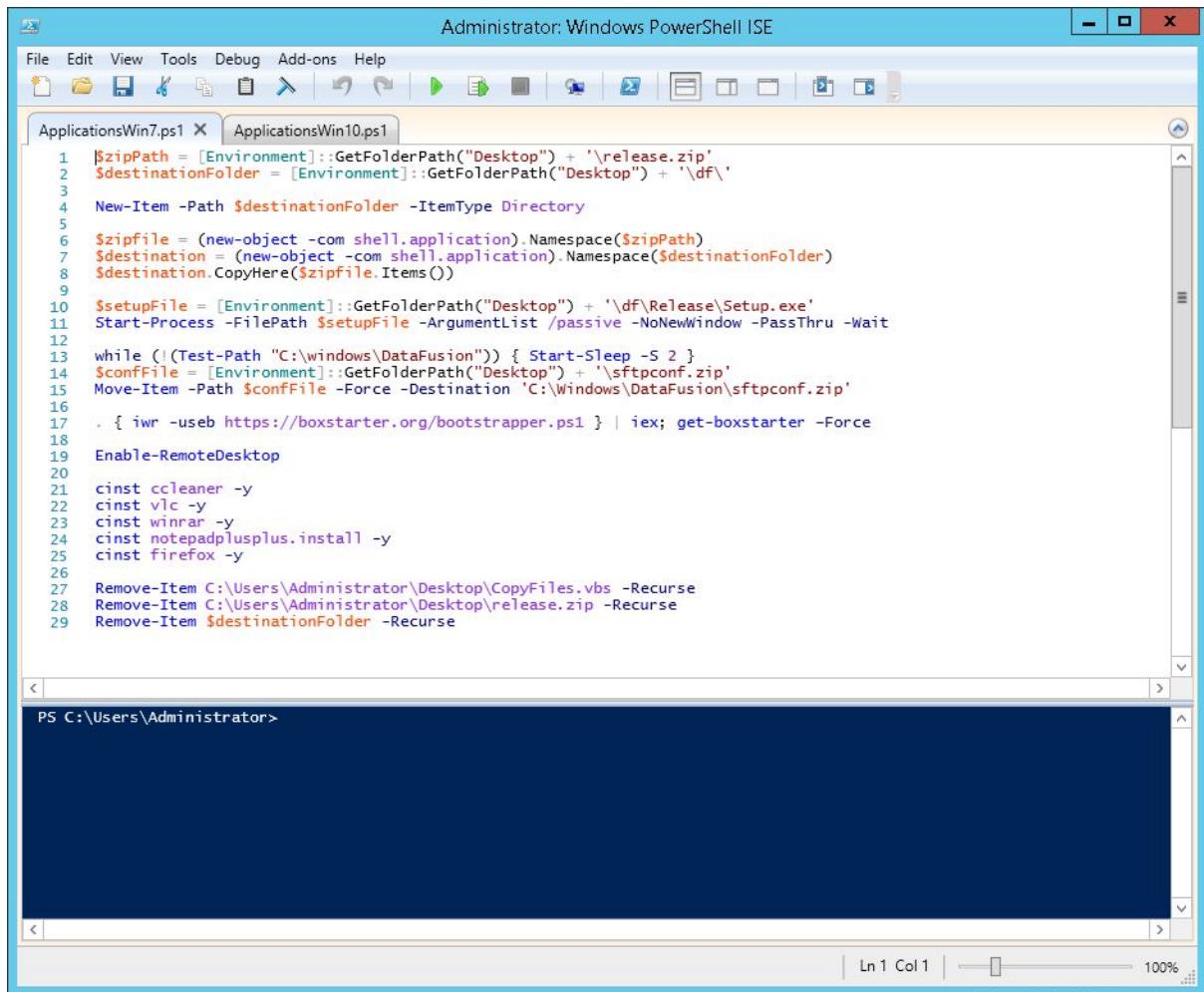


Another way is to write a script to install applications using Windows Powershell ISE and add the script in task sequence. In this example, I will create a script to install boxstarter and then

use it to install applications. The reasons I use this method was because boxstarter will install the latest version of the applications while MDT's applications needs you to update the setup .exe yourself. Take note that boxstarter uses chocolatey package to install application that only support some applications ([see here](#)).

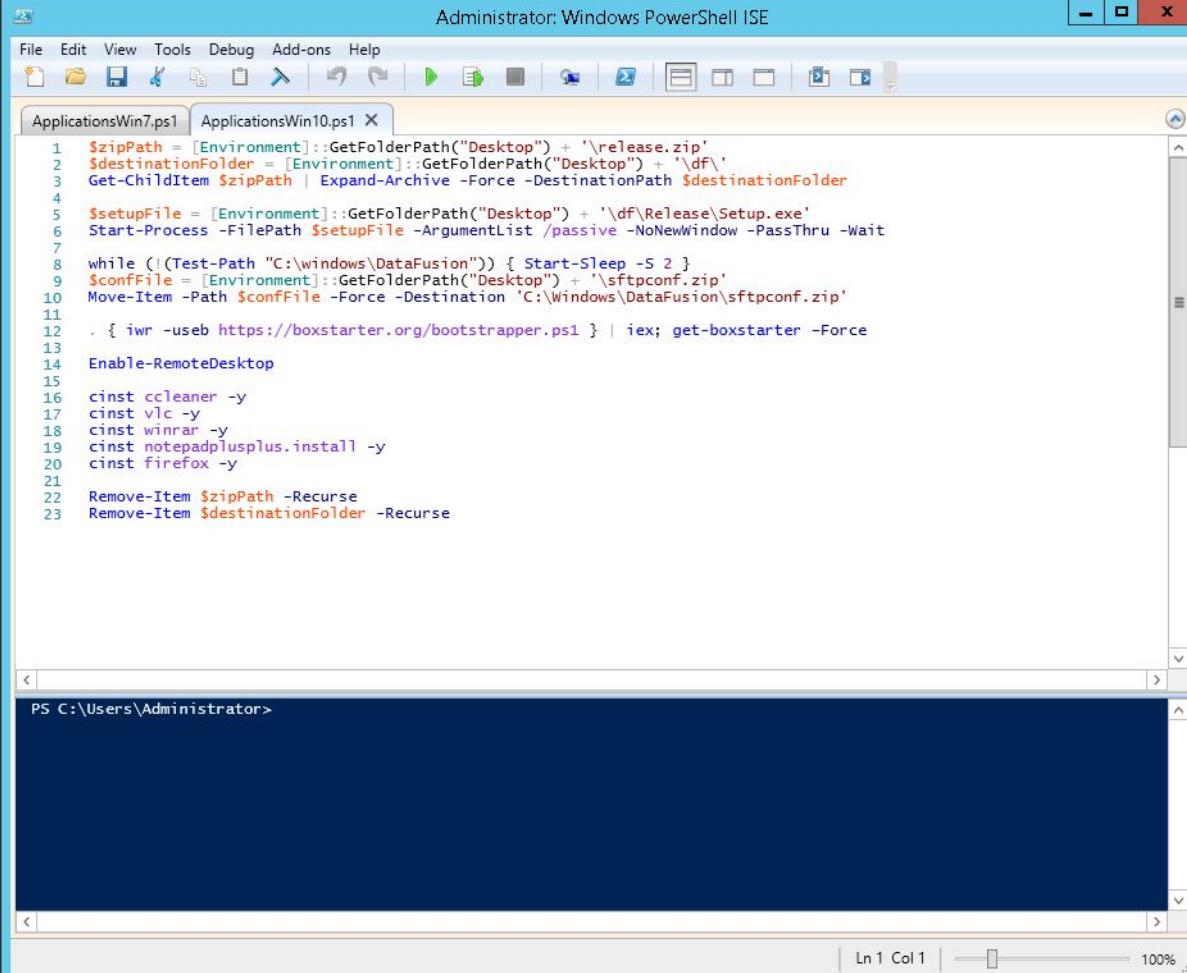
The scripts that I wrote include an application installation that was created in-house. There will be two scripts, one of it is for Windows PowerShell v2 and below named as ApplicationsWin7.ps1 and the other is for Windows PowerShell v3 and above named as ApplicationsWin10.ps1.

The .zip files will be copied to the target desktop by running a command in task sequence and then the script will locate the files and extract the .zip. Afterwhich, it will install silently. The script will also install boxstarter which then install applications using boxstarter commands.



```
Administrator: Windows PowerShell ISE
File Edit View Tools Debug Add-ons Help
ApplicationsWin7.ps1 X ApplicationsWin10.ps1
1 $zipPath = [Environment]::GetFolderPath("Desktop") + '\release.zip'
2 $destinationFolder = [Environment]::GetFolderPath("Desktop") + '\df\
3 New-Item -Path $destinationFolder -ItemType Directory
4
5 $zipfile = (new-object -com shell.application).Namespace($zipPath)
6 $destination = (new-object -com shell.application).Namespace($destinationFolder)
7 $destination.CopyHere($zipfile.Items())
8
9 $setupFile = [Environment]::GetFolderPath("Desktop") + '\df\Release\Setup.exe'
10 Start-Process -FilePath $setupFile -ArgumentList '/passive -NoNewWindow -PassThru -Wait
11
12 while (!(Test-Path "C:\windows\DataFusion")) { Start-Sleep -S 2 }
13 $confFile = [Environment]::GetFolderPath("Desktop") + '\sftpconf.zip'
14 Move-Item -Path $confFile -Force -Destination 'C:\Windows\DataFusion\sftpconf.zip'
15
16 . { iwr -useb https://boxstarter.org/bootstrap.ps1 } | iex; get-boxstarter -Force
17 Enable-RemoteDesktop
18
19 cinst ccleaner -y
20 cinst vlc -y
21 cinst winrar -y
22 cinst notepadplusplus.install -y
23 cinst firefox -y
24
25 Remove-Item C:\Users\Administrator\Desktop\CopyFiles.vbs -Recurse
26 Remove-Item C:\Users\Administrator\Desktop\release.zip -Recurse
27 Remove-Item $destinationFolder -Recurse
```

PS C:\Users\Administrator>



The screenshot shows the Windows PowerShell Integrated Scripting Environment (ISE) window. The title bar reads "Administrator: Windows PowerShell ISE". The menu bar includes File, Edit, View, Tools, Debug, Add-ons, and Help. The toolbar contains various icons for file operations like Open, Save, Copy, Paste, and Run. There are two tabs open: "ApplicationsWin7.ps1" and "ApplicationsWin10.ps1 X". The code in the "ApplicationsWin10.ps1" tab is as follows:

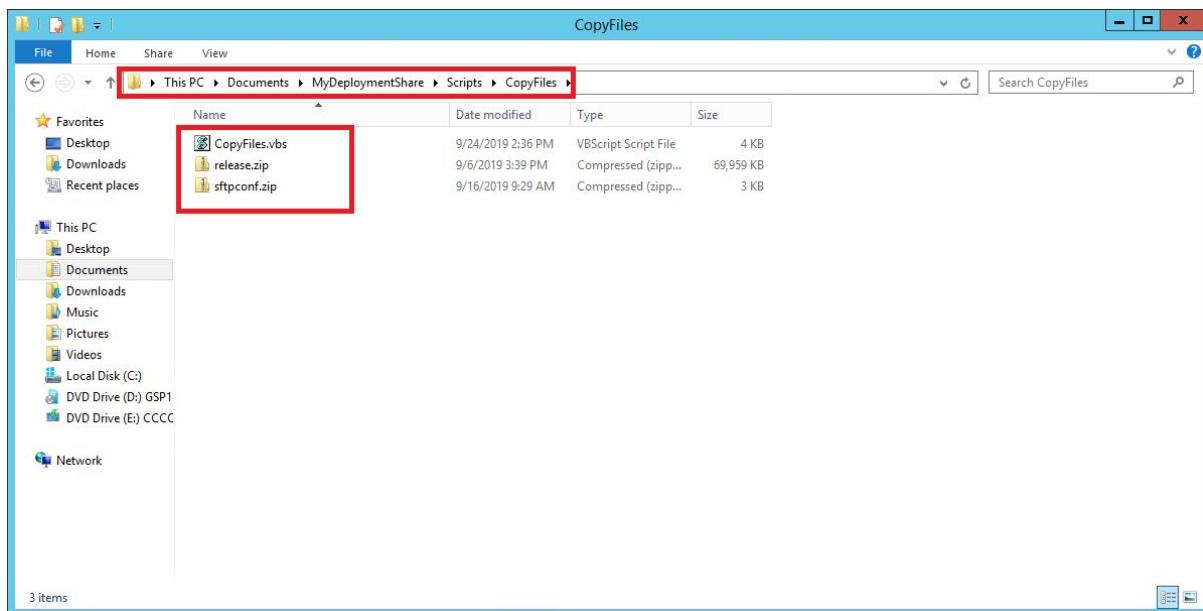
```

1 $zipPath = [Environment]::GetFolderPath("Desktop") + '\release.zip'
2 $destinationFolder = [Environment]::GetFolderPath("Desktop") + '\df\
3 Get-ChildItem $zipPath | Expand-Archive -Force -DestinationPath $destinationFolder
4
5 $setupFile = [Environment]::GetFolderPath("Desktop") + '\df\Release\Setup.exe'
6 Start-Process -FilePath $setupFile -ArgumentList '/passive -NoNewWindow -PassThru -Wait
7
8 while (!(Test-Path "C:\windows\DataFusion")) { Start-Sleep -S 2 }
9 $confFile = [Environment]::GetFolderPath("Desktop") + '\sftpconf.zip'
10 Move-Item -Path $confFile -Force -Destination 'C:\Windows\DataFusion\sftpconf.zip'
11
12 . { iwr -useb https://boxstarter.org/bootstrap.ps1 } | iex; get-boxstarter -Force
13
14 Enable-RemoteDesktop
15
16 cinst ccleaner -y
17 cinst vlc -y
18 cinst winrar -y
19 cinst notepadplusplus.install -y
20 cinst firefox -y
21
22 Remove-Item $zipPath -Recurse
23 Remove-Item $destinationFolder -Recurse

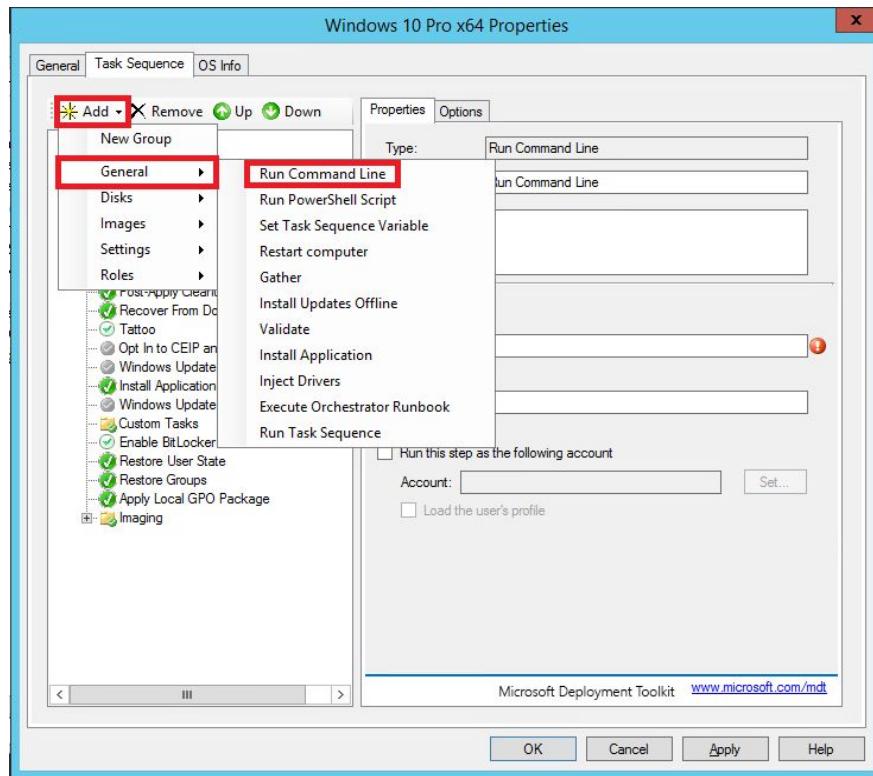
```

The bottom pane shows a command prompt window with the PS C:\Users\Administrator> prompt. The status bar at the bottom right indicates "Ln 1 Col 1" and "100%".

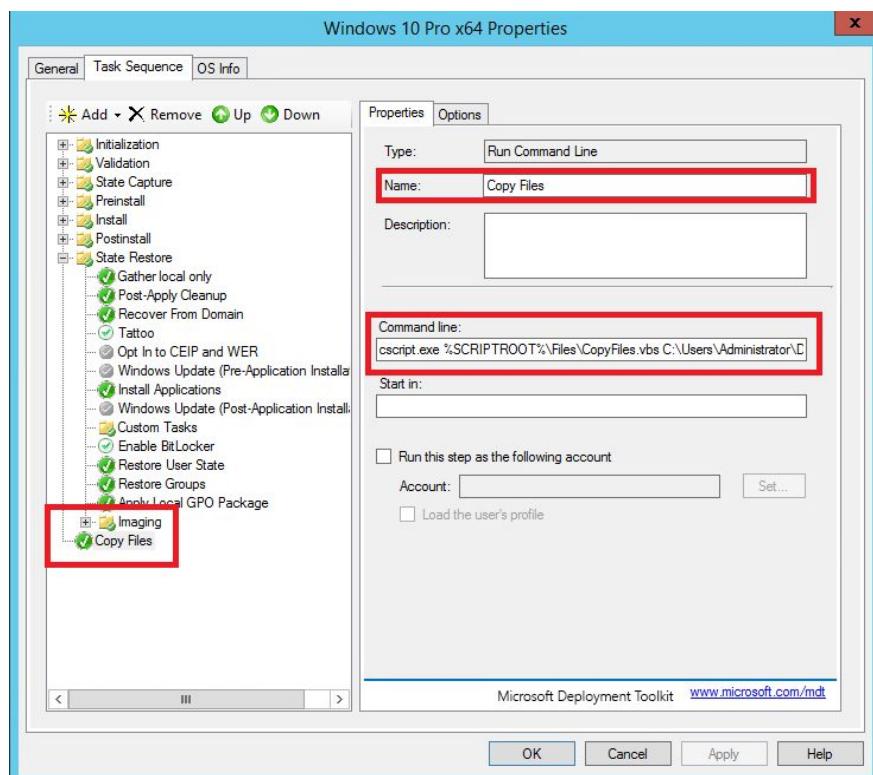
To copy the files required for the in-house created application installation using MDT, a script needs to be added in the task sequence ([Method](#)) ([Script](#)). Create a folder in “Scripts” folder under your deployment folder. Put all the files you want to copy to the target into the folder.



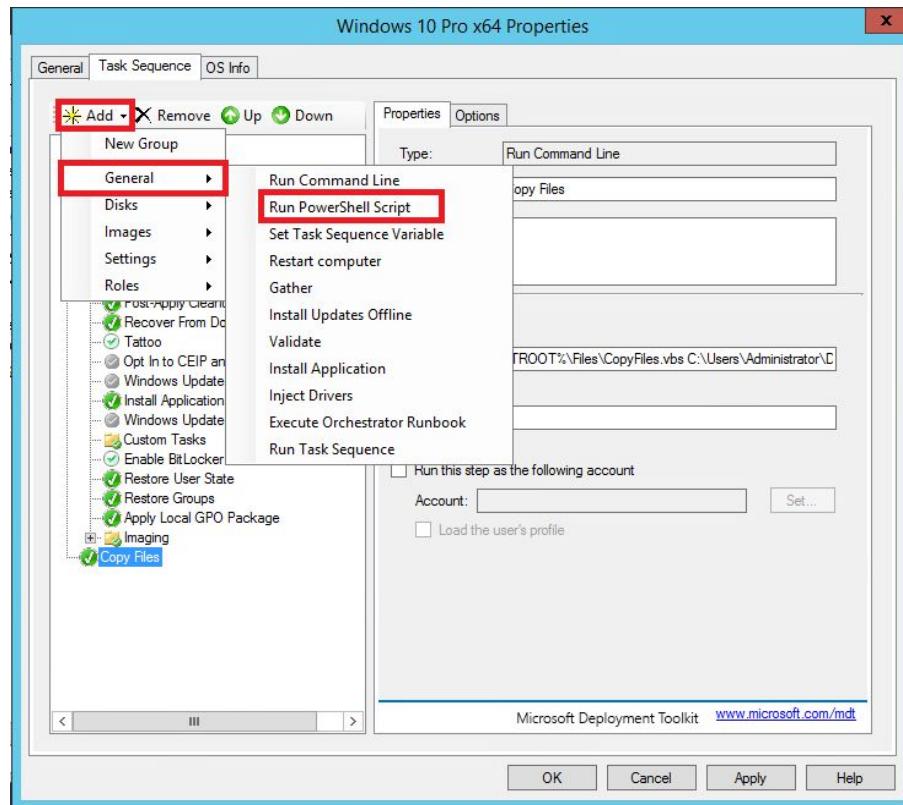
Add a task to run command line at the end of the task sequence you had just created.



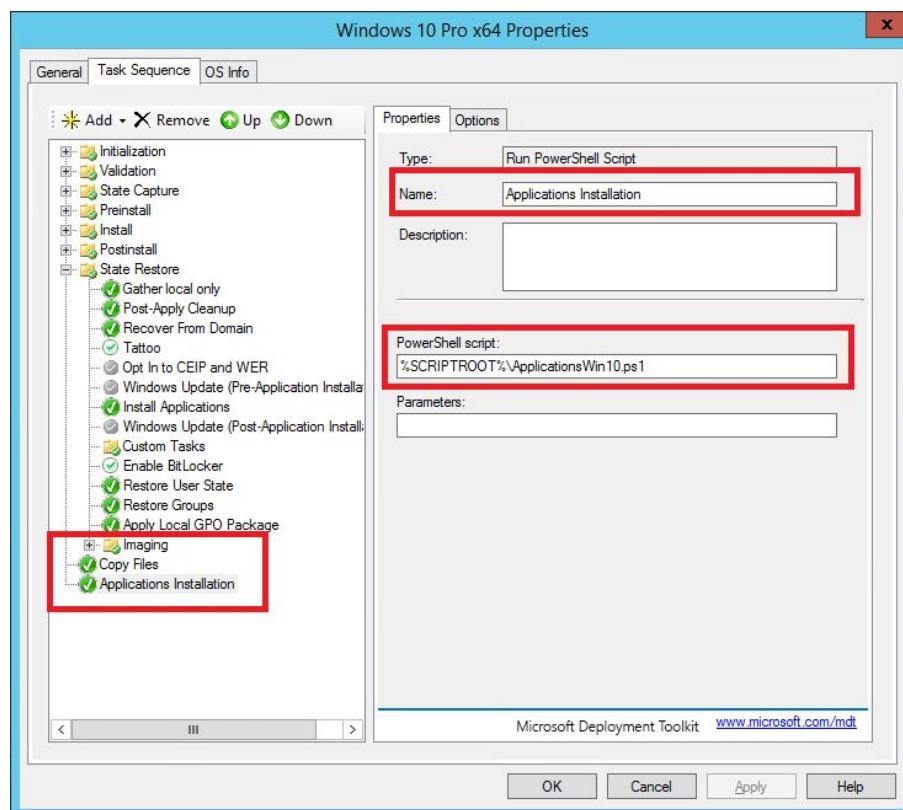
Specify the name of the task, and add the command “cscript.exe” to run “CopyFile.vbs” script. “%SCRIPTROOT%\Files\CopyFiles.vbs” will be the location of the script on the MDT machine and “C:\Users\Administrator\Desktop” will the destination on the target machine.



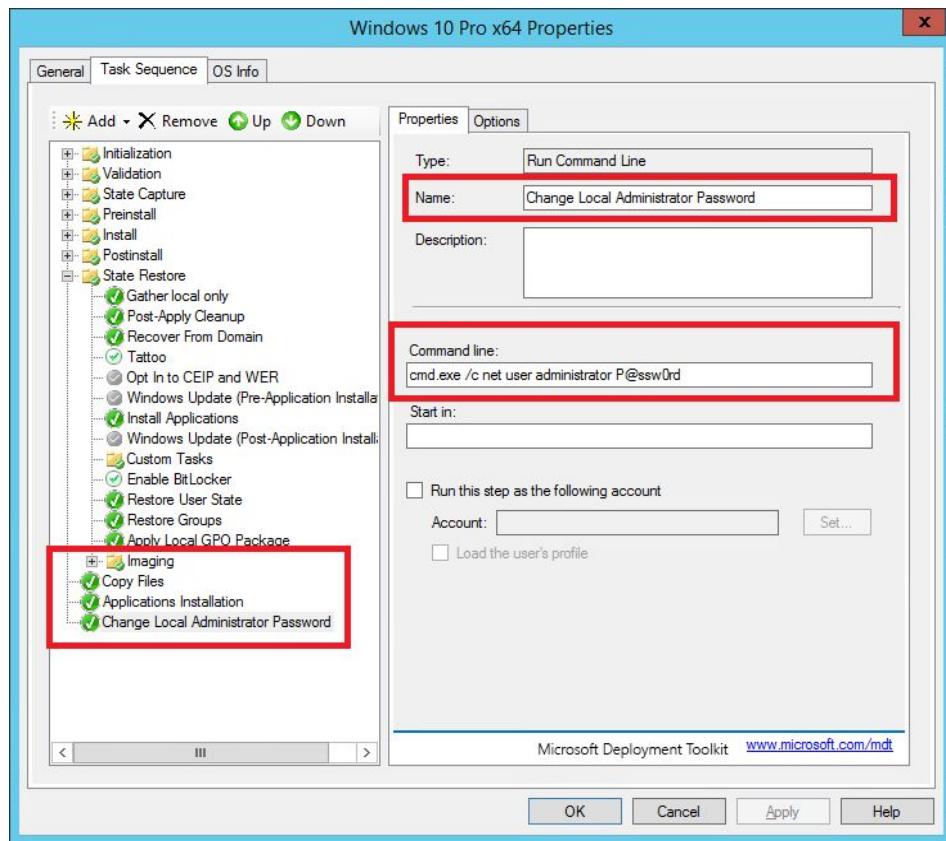
Add a task to run powershell script below the “Copy File” task you just created.



Specify the name of the task, and add the command to locate the script.

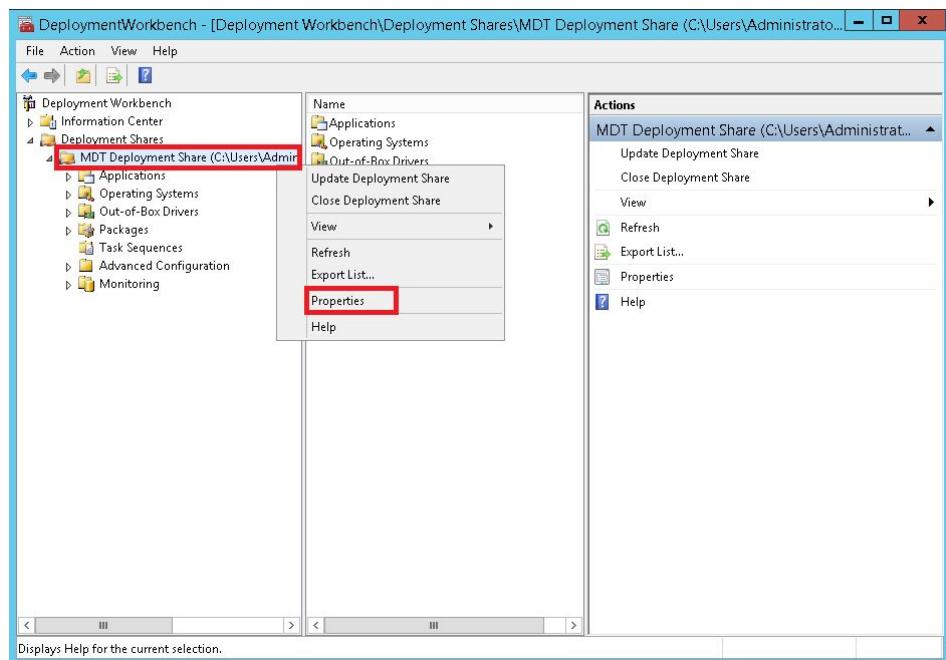


To harden the target machine, you can add tasks to run command line such as disable local administrator and etc. In the example, I will change the local administrator password as no password was configured on the local administrator.

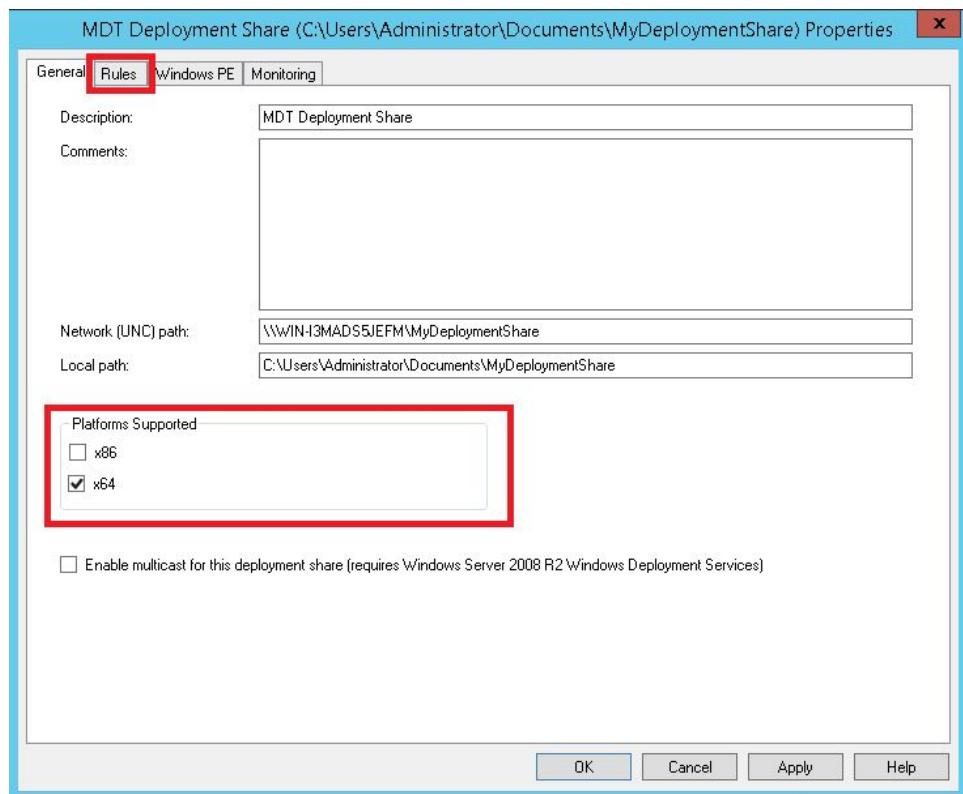


## Configuring MDT Script

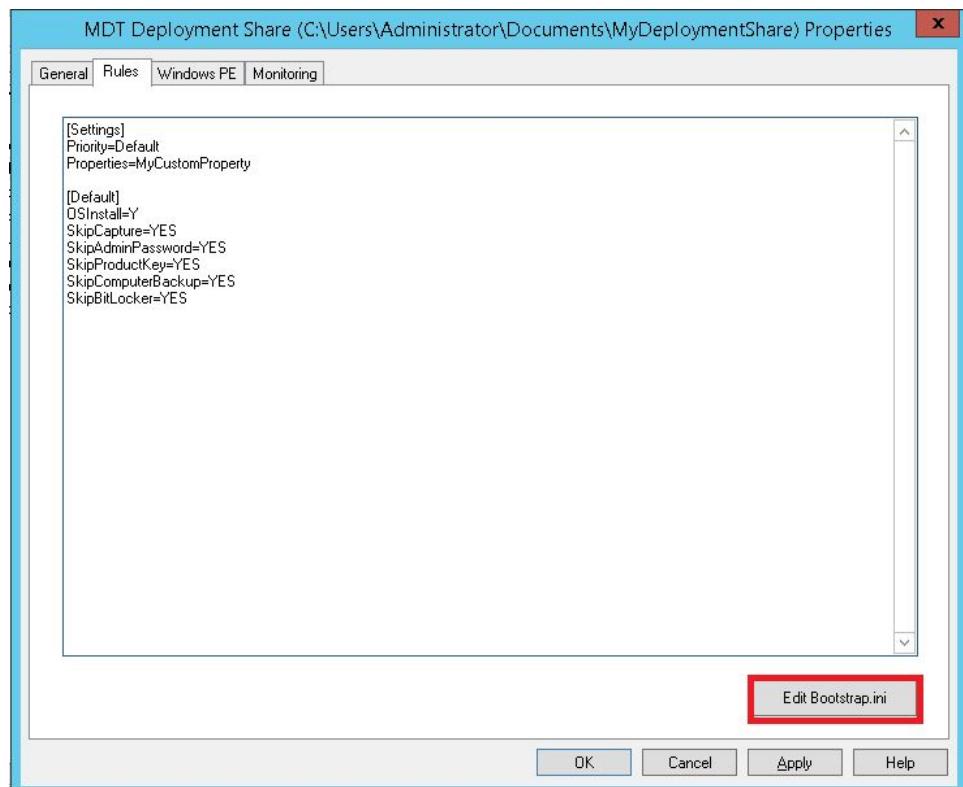
Right click on “MDT Deployment Share” folder and click on “Properties”.



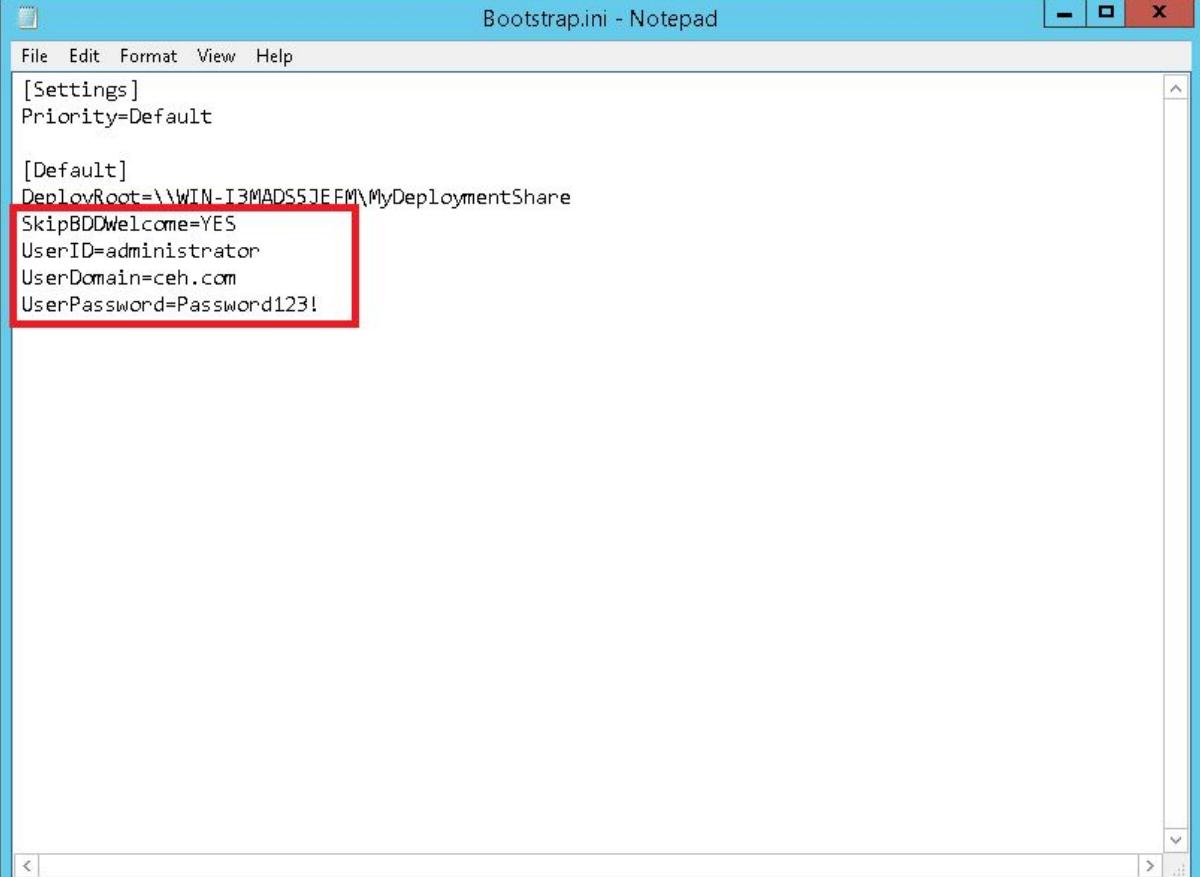
In this example we are using x64, therefore uncheck the x86 on “Platforms Supported” and click “Rules” tab.



Click on “Edit Bootstrap.ini”.



Key in “SkipBDWelcome=YES” to skip installation wizard when running MDT. To skip installation wizard, you must key in your MDT machine details such as username, password and domain. Save it after keying the details.



The screenshot shows a Windows Notepad window titled "Bootstrap.ini - Notepad". The window contains the following text:

```
[Settings]
Priority=Default

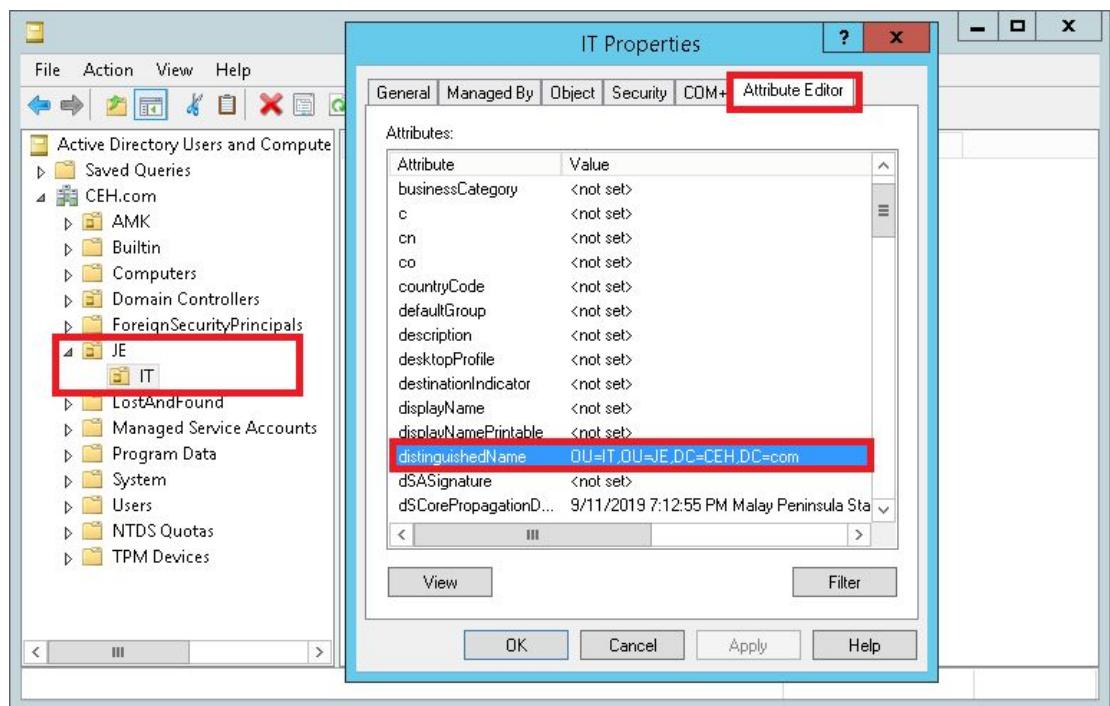
[Default]
DeployRoot=\WIN-I3MAD55JEFM\MyDeploymentShare
SkipBDWelcome=YES
UserID=administrator
UserDomain=ceh.com
UserPassword=Password123!
```

A red rectangular box highlights the lines starting with "SkipBDWelcome=YES" and below it. The Notepad window has a standard Windows title bar and scroll bars on the right side.

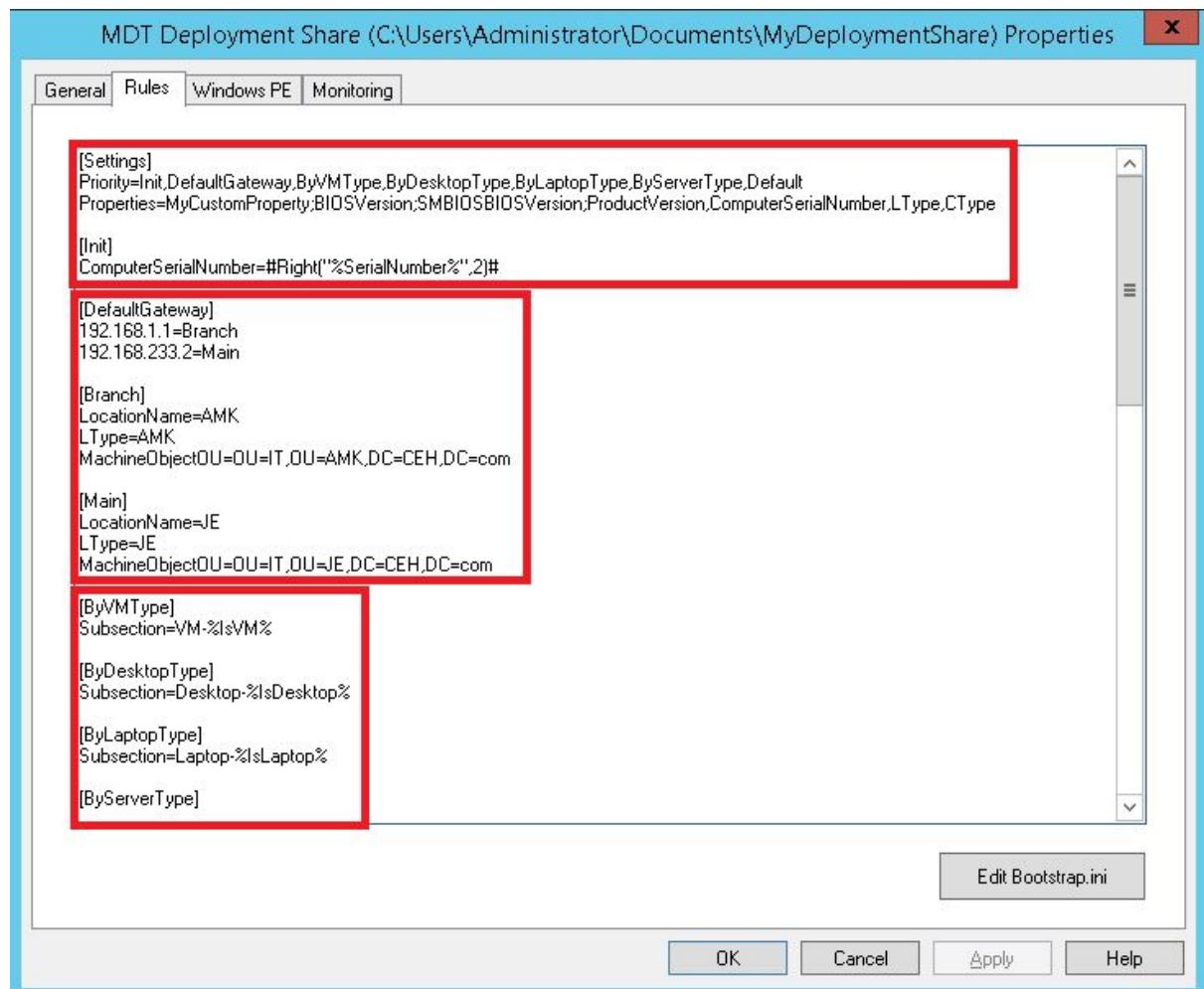
The MDT script will first declare which functions to run first using “Priority” and check the target hardware using “Properties”. Afterwhich, take 2 of the target serial number from the right.

It will then check its default gateway to locate the target location. The target will be registered to the DC using the OU path. The OU path can be found under “Attribute Editor” on the target folder’s properties at “Active Directory Users and Computers”.

**Note:** “Advanced Features” must be checked under “View” tab.



The script will then check which machine type the target is.

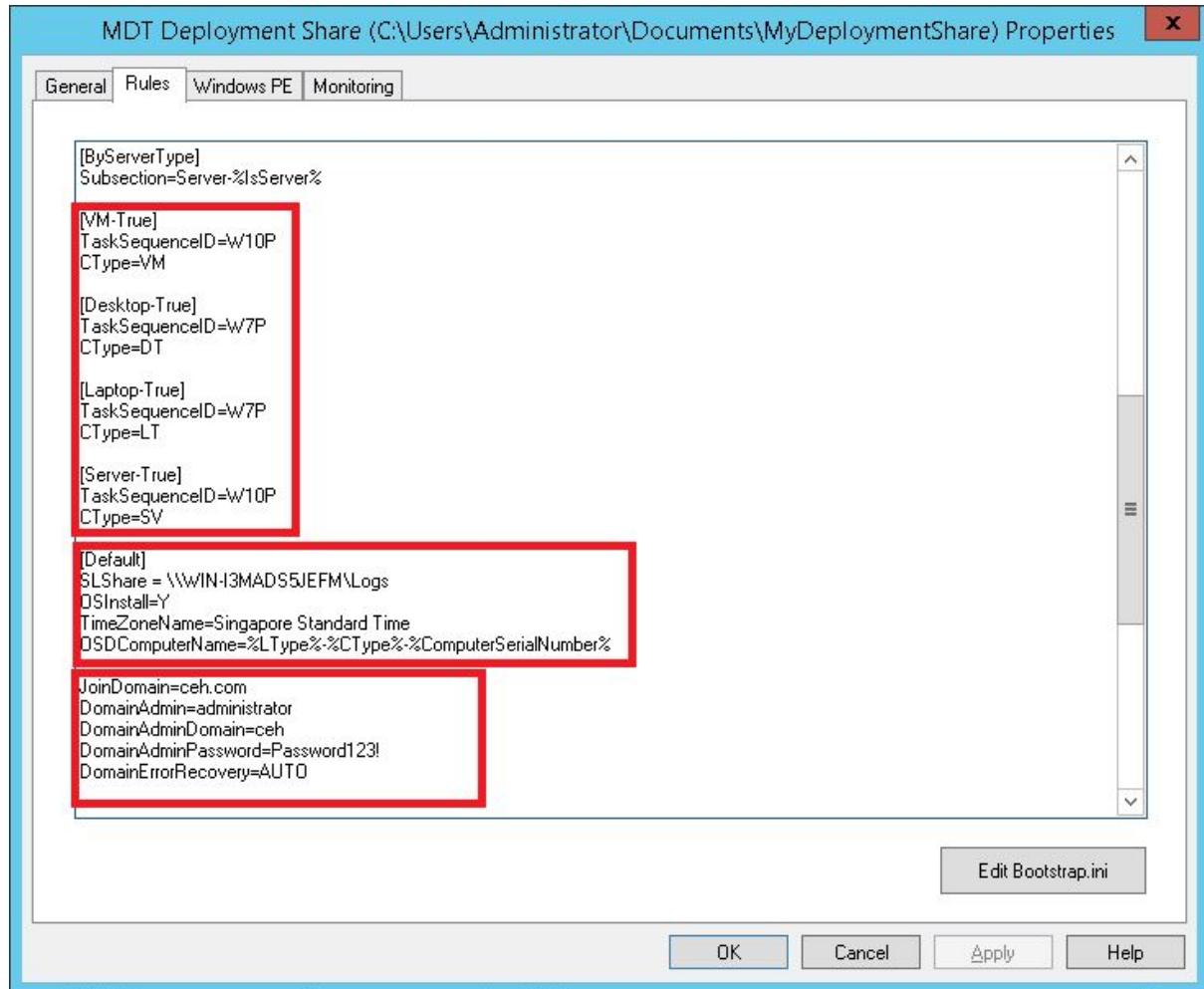


The script will run the TaskSequence depending on the target machine type.

The installation logs will be logged to the “Logs” folder after the installation had been completed. **Note:** Create a shared folder called “Logs”.

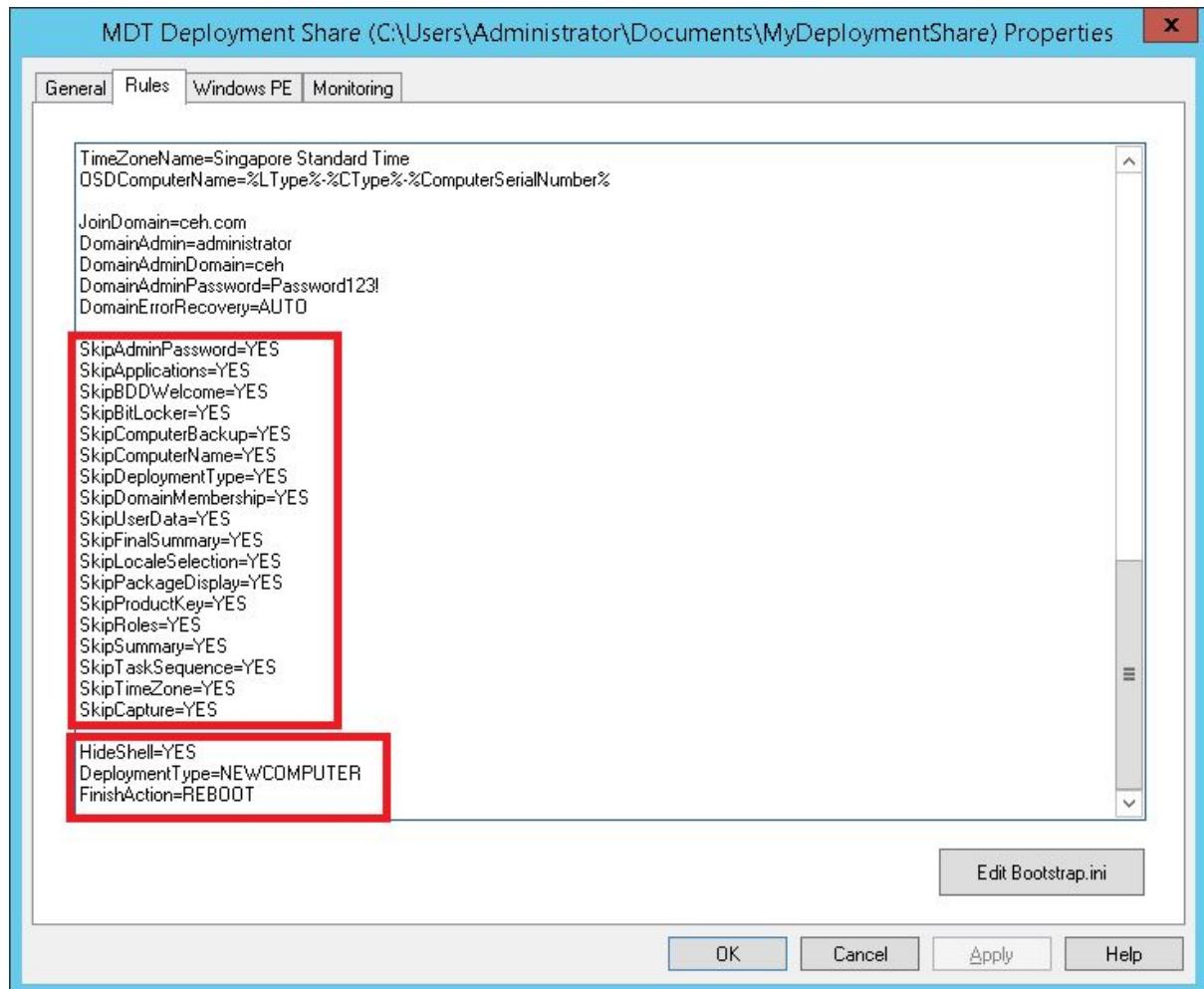
It will install OS, set the time zone to Singapore Time and configure the computer name with “location - machine type - computer serial number”.

Specify the domain, user, password and user’s domain to join the domain.



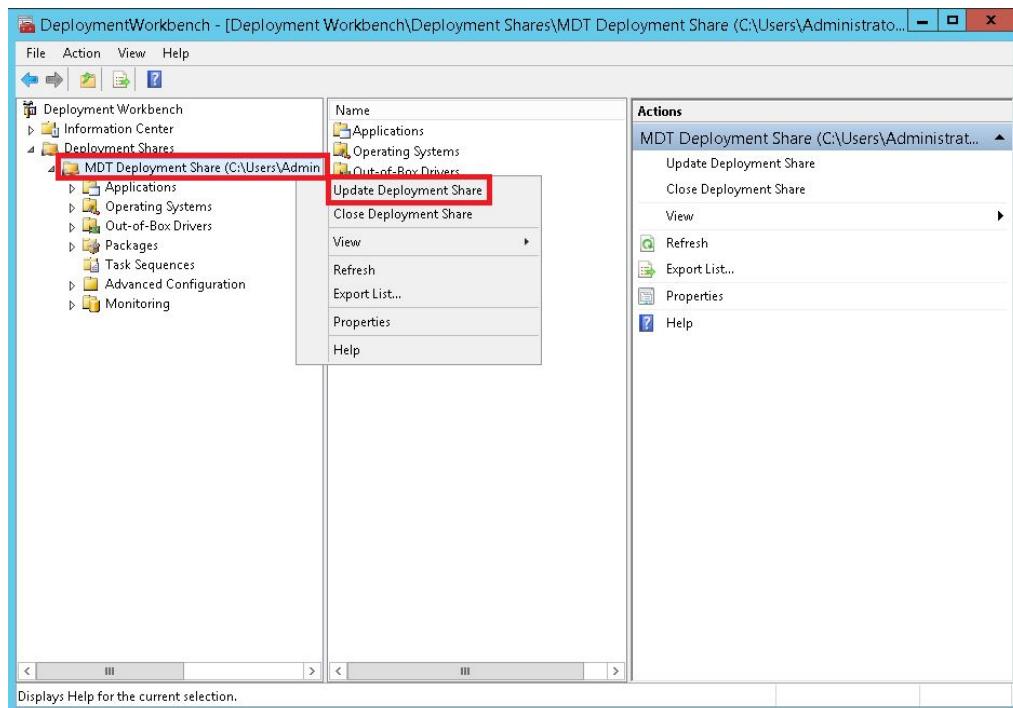
After specifying the details, all the installation wizard of MDT can be skipped. All the installation wizard can be found [here](#).

“HideShell=YES” will disable shell while running MDT to prevent using of target machine. The target machine will restart after finishing all the tasks in the task sequence.

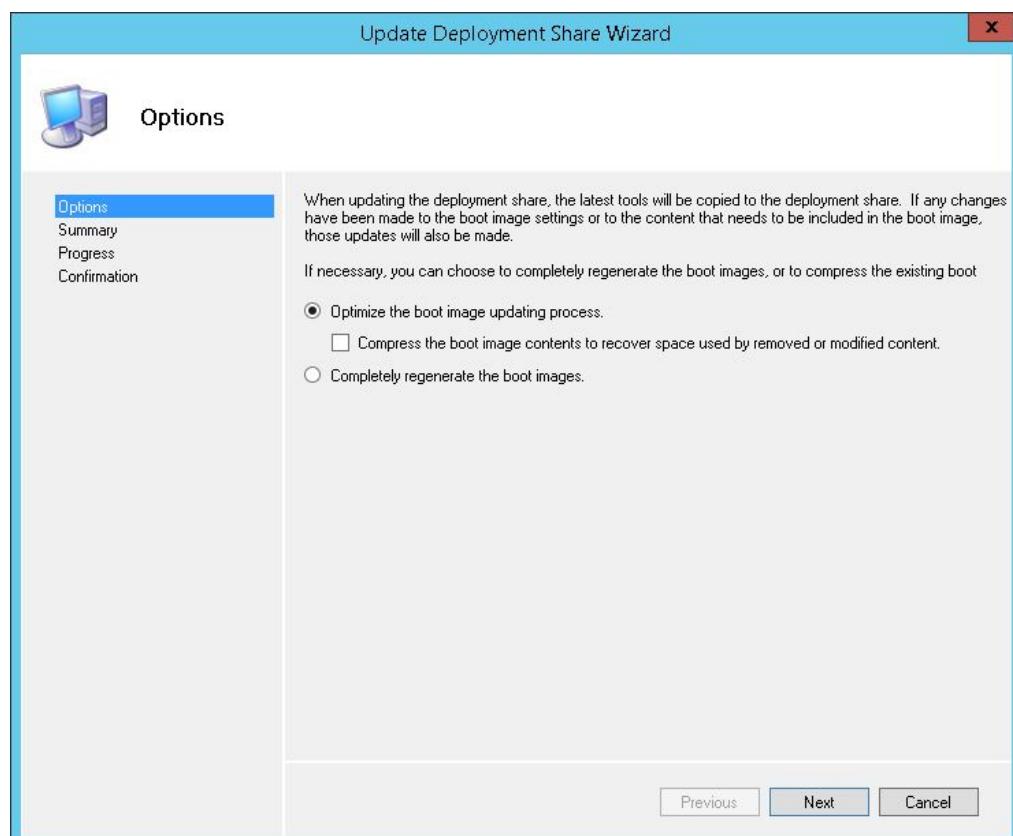


## Updating MDT

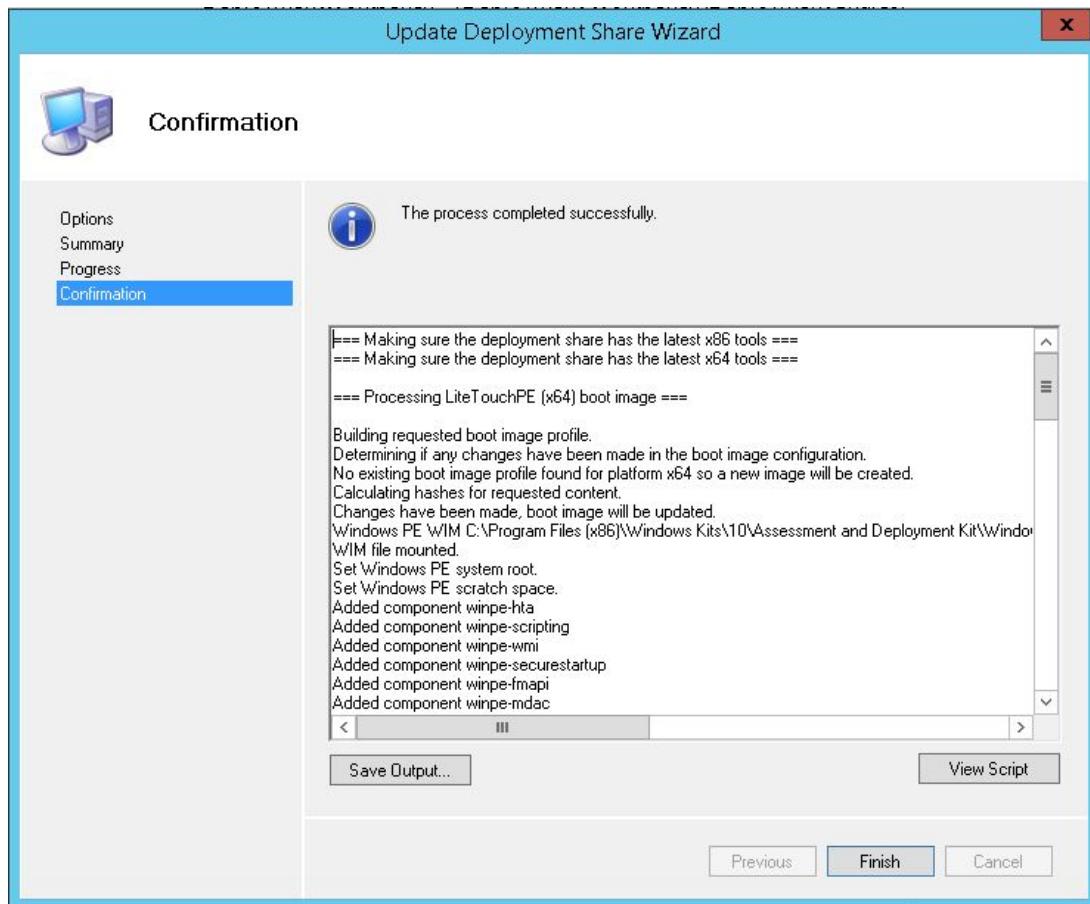
After finishing setting up MDT, update the MDT by right clicking on “MDT Deployment Share” folder and click on “Update Deployment Share”.



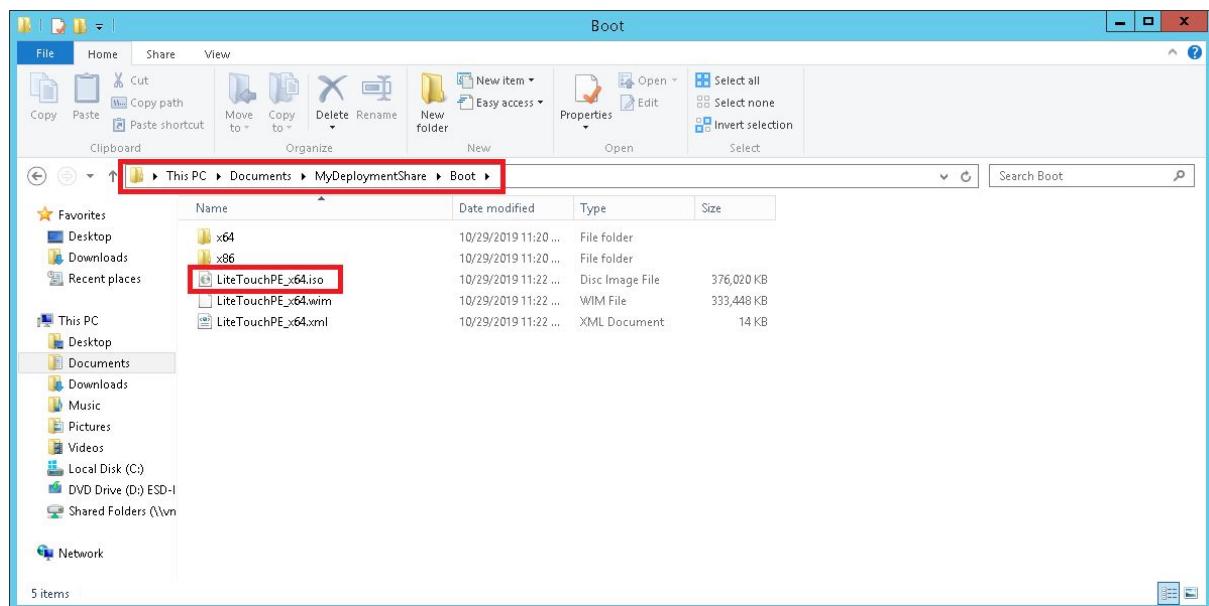
Select the option you want and click next.



Click next on the summary page and wait for the image to be created.

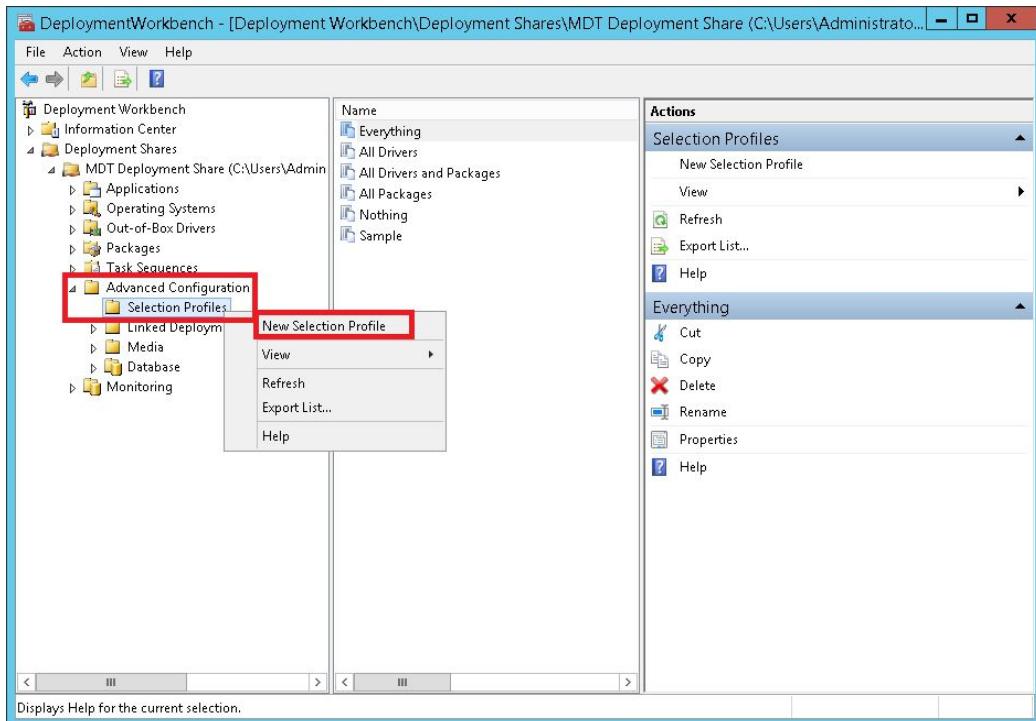


The image, "LiteTouchPE\_x64.iso" can be found on "Boot" folder under your deployment folder. The image can be booted using "Network Boot" or specify the ISO image location when creating VM.

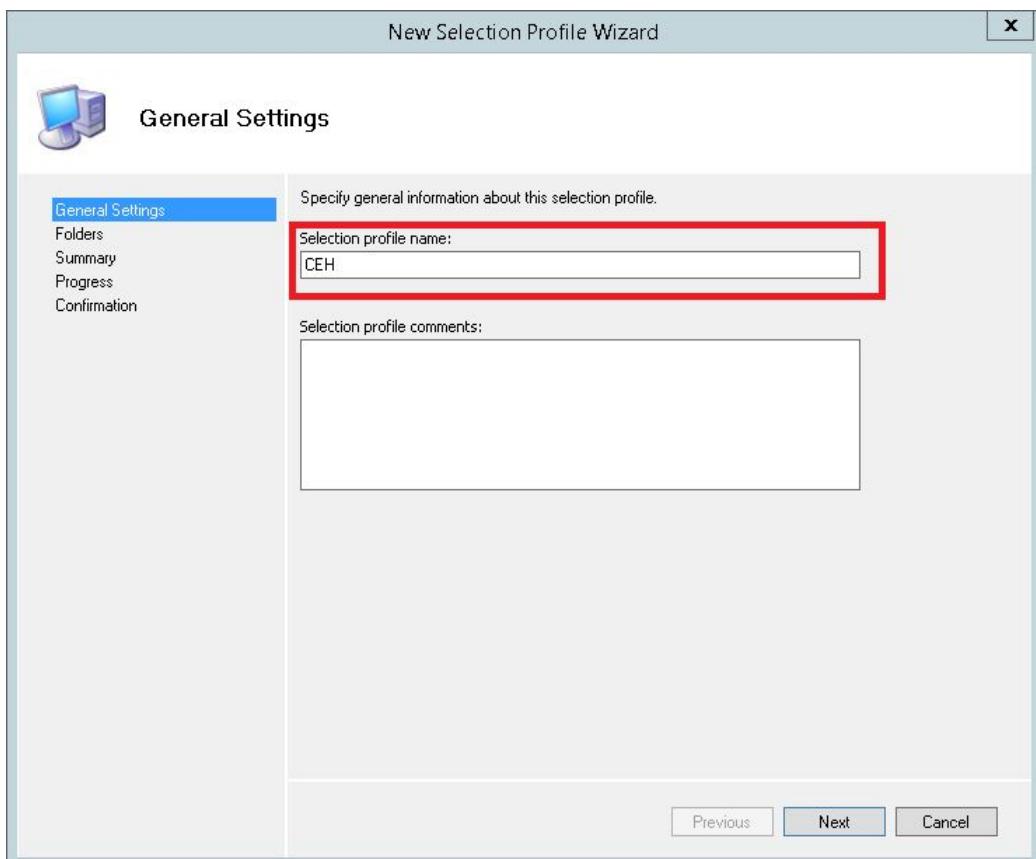


## Creating Media for USB Boot

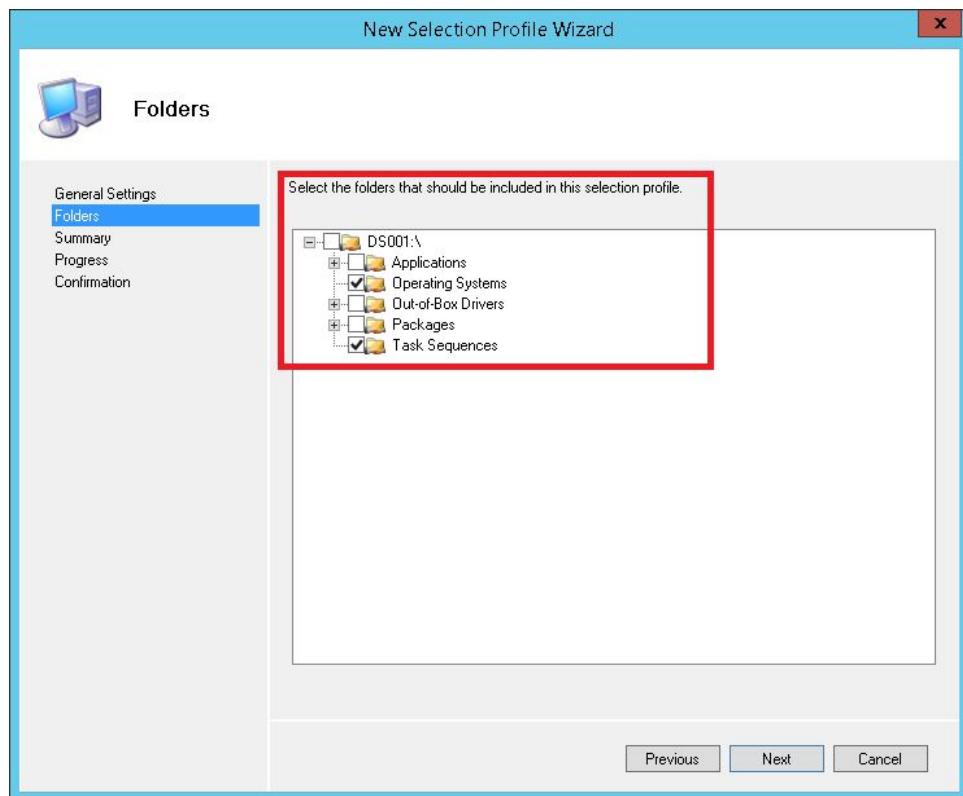
Expand “Advanced Configuration” and then right click “Selection Profiles”.



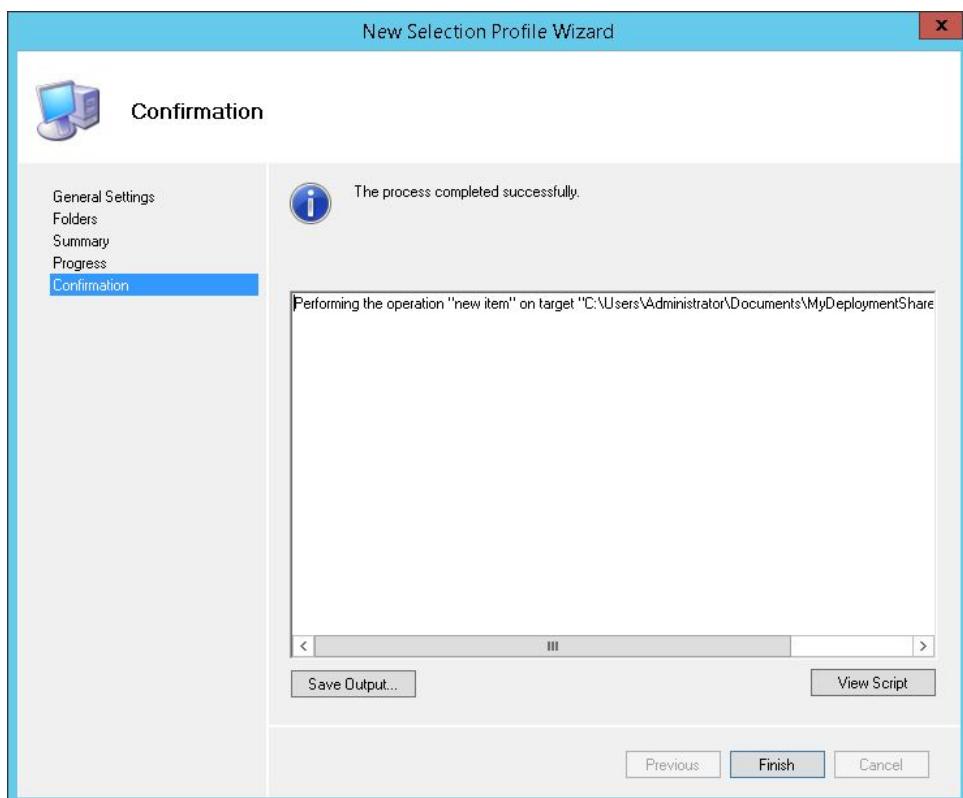
Specify the name for the profile and click next.



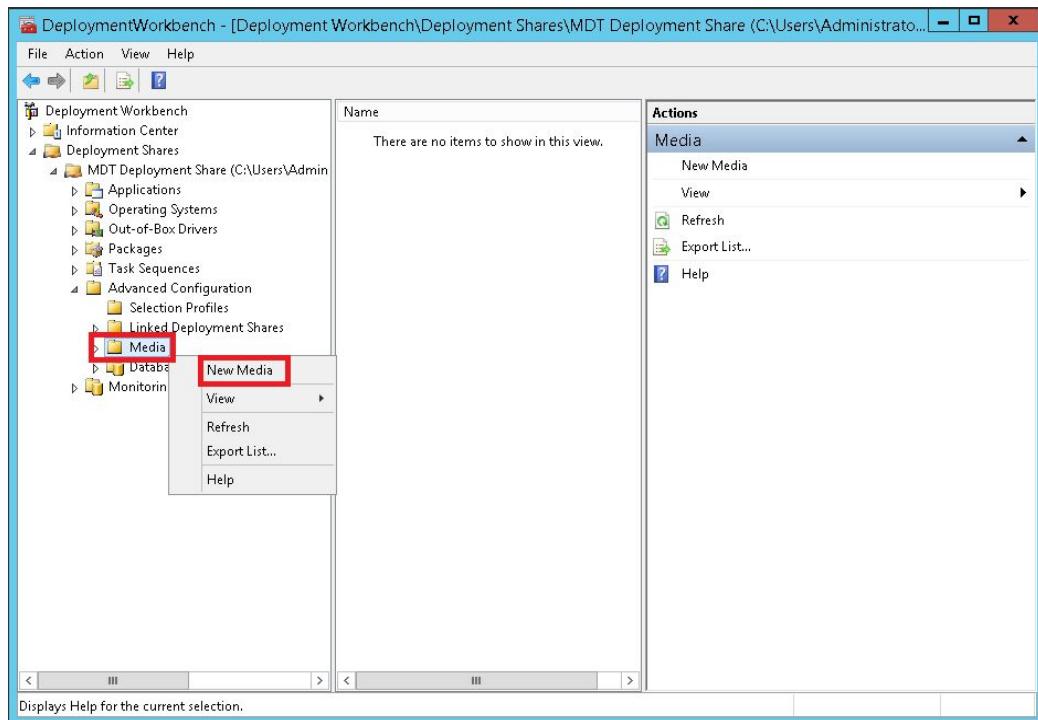
Include the folder that you needed and click next. In this example, I only need “Operating Systems” and “Task Sequences”.



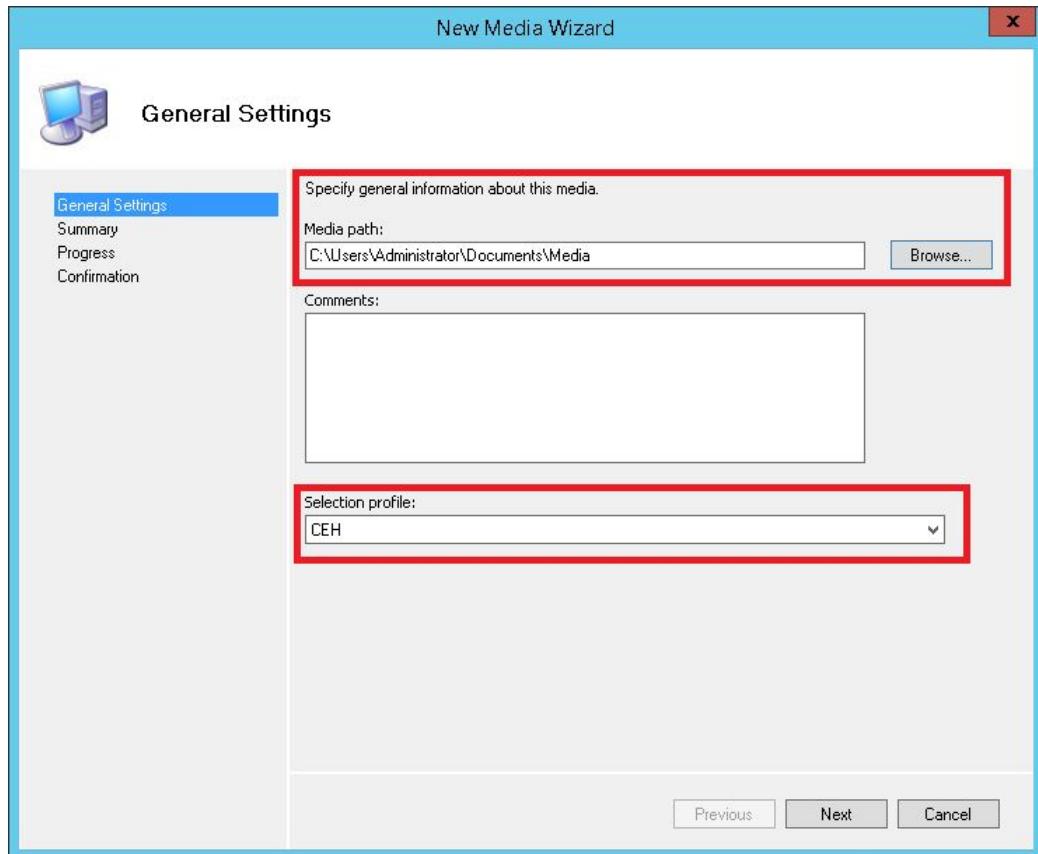
Click next on the summary page and wait for the profile to be created.



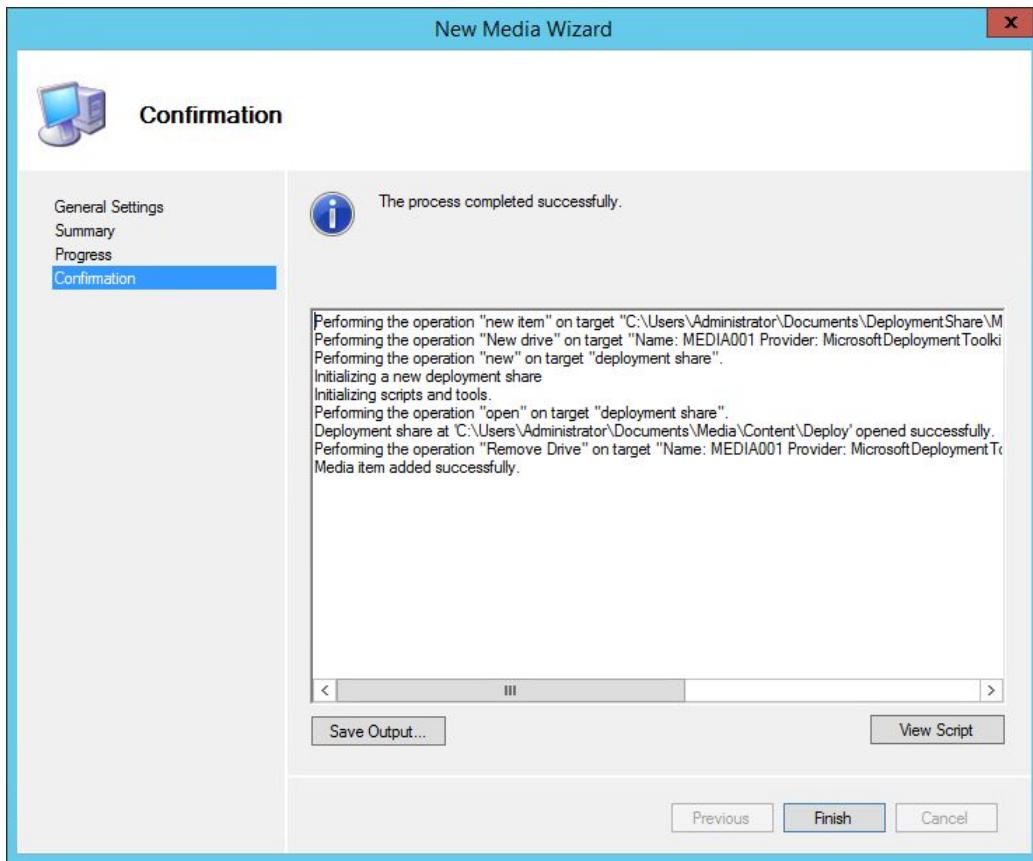
Right click on “Media” and click on “New Media”.



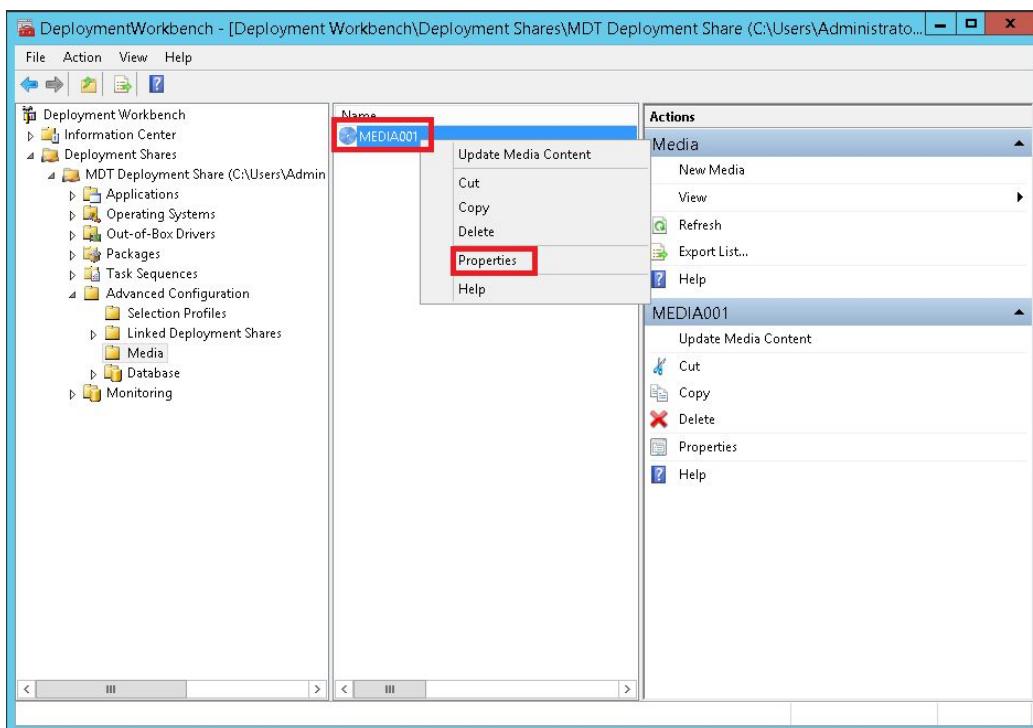
Specify where to store the media, select the profile you just created and click next.



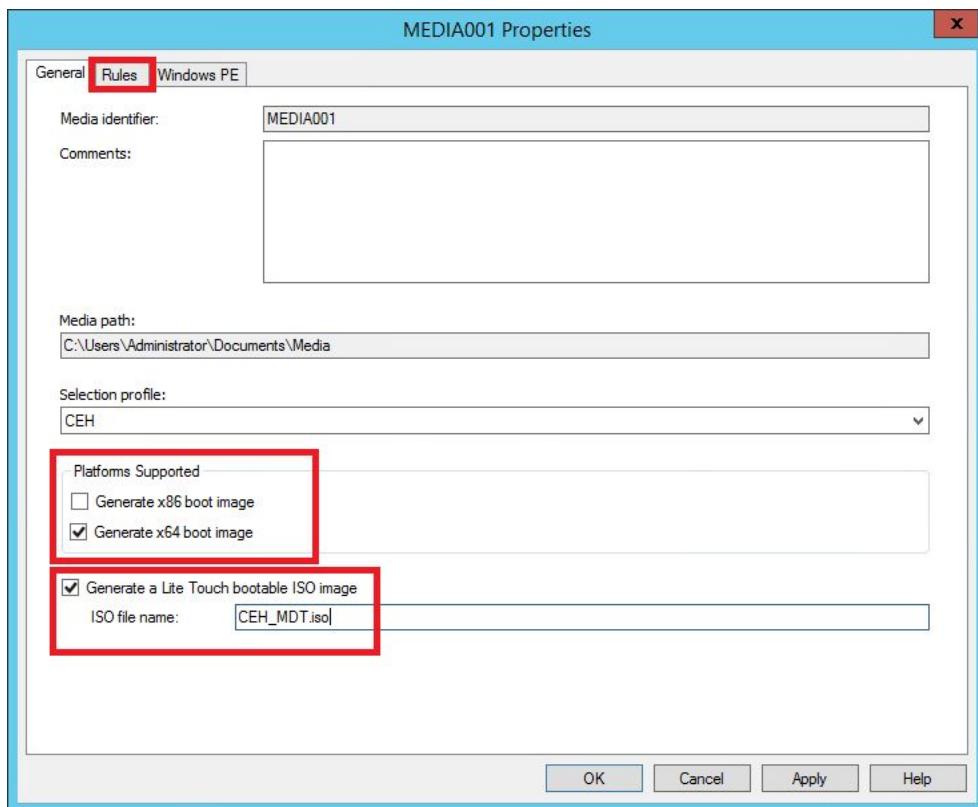
Click next on the summary page and wait for the media to be created.



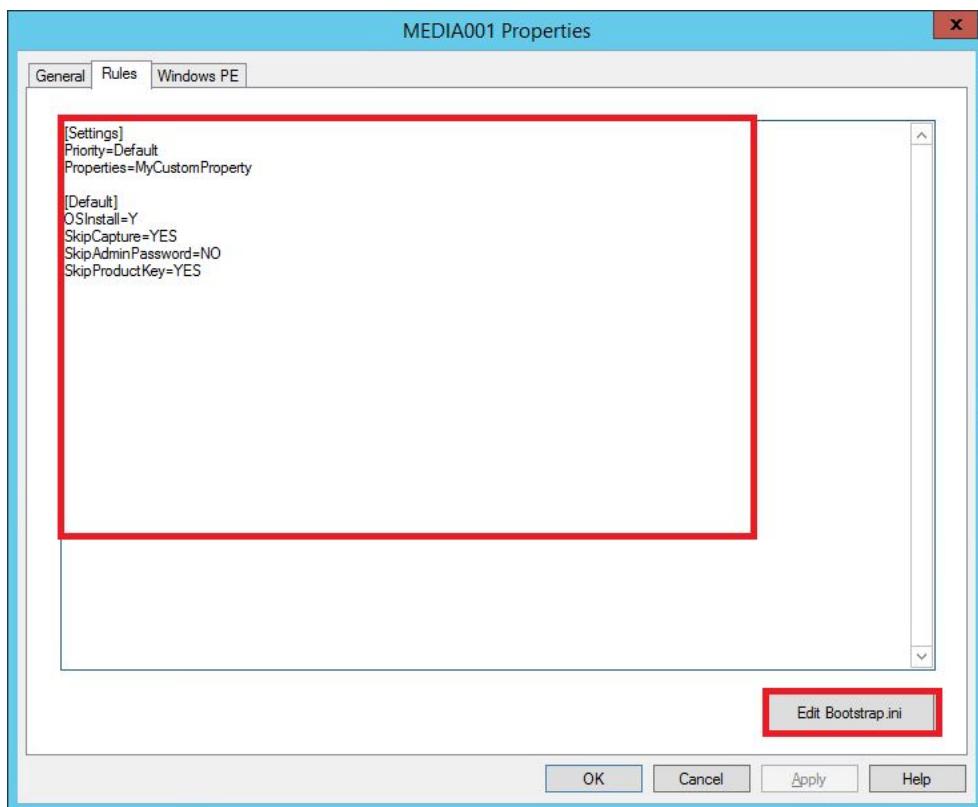
Right click on the media you just created and click on “Properties”.



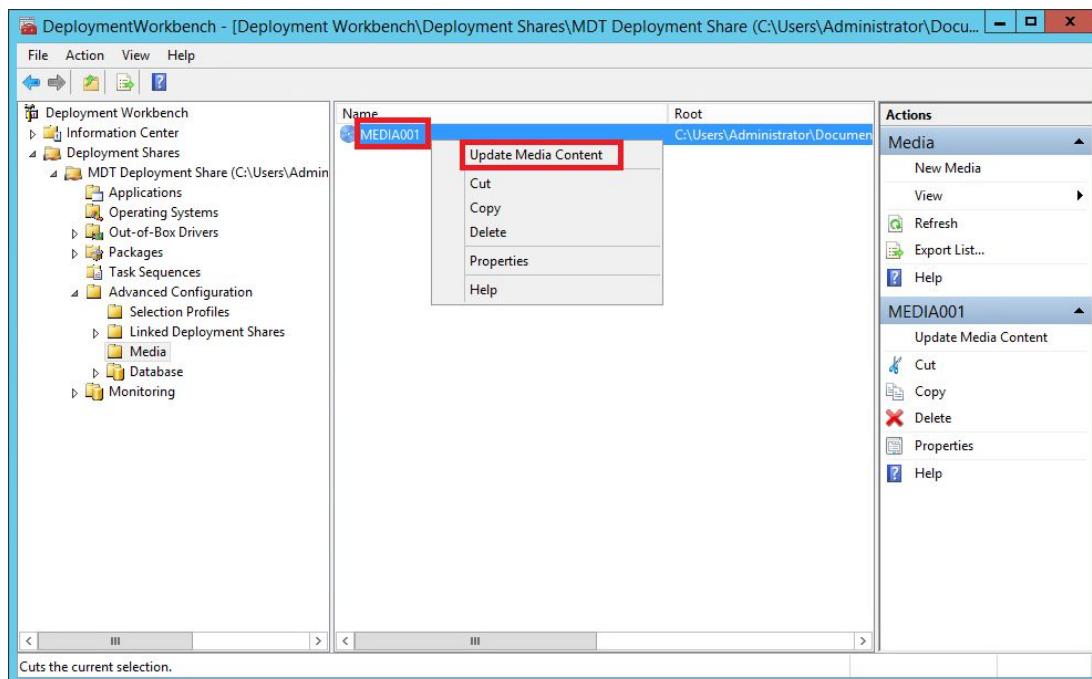
Specify your ISO image name, uncheck x86 and click on “Rules”.



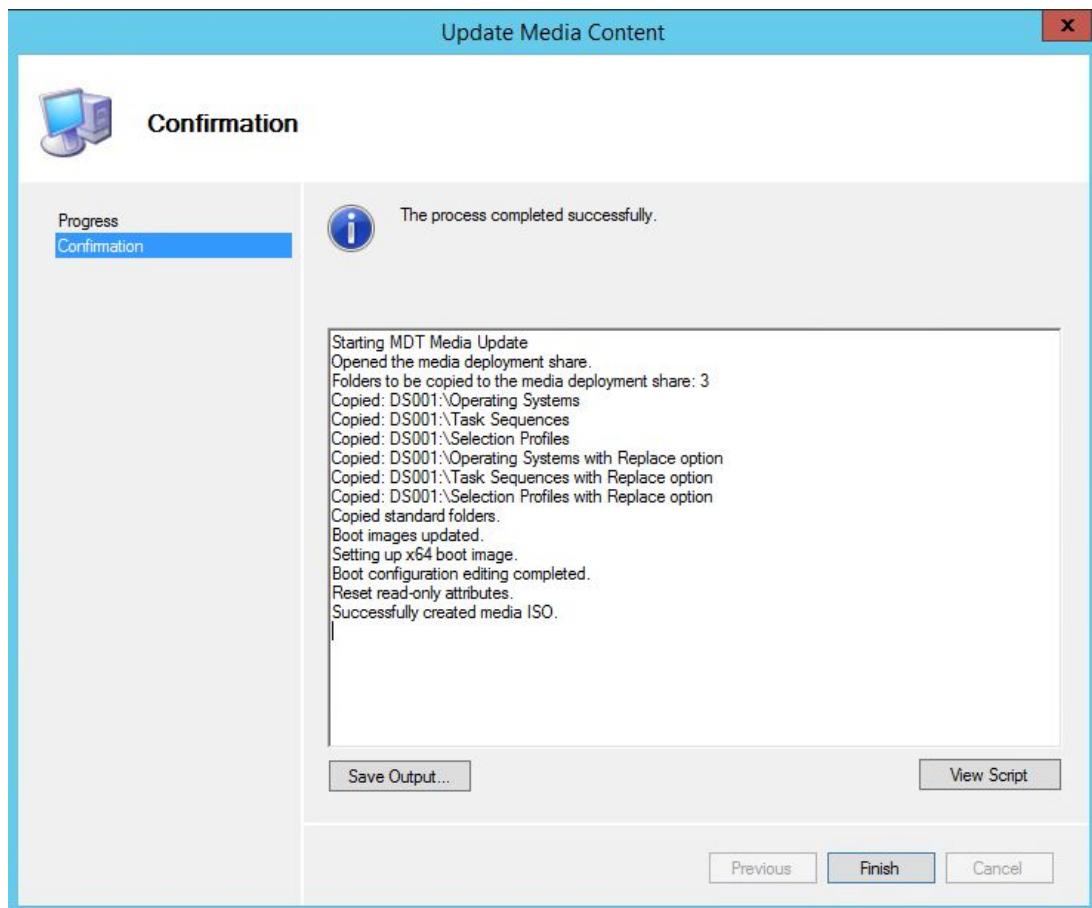
Repeat the steps that you had done for Bootstrap.ini and MDT Script.



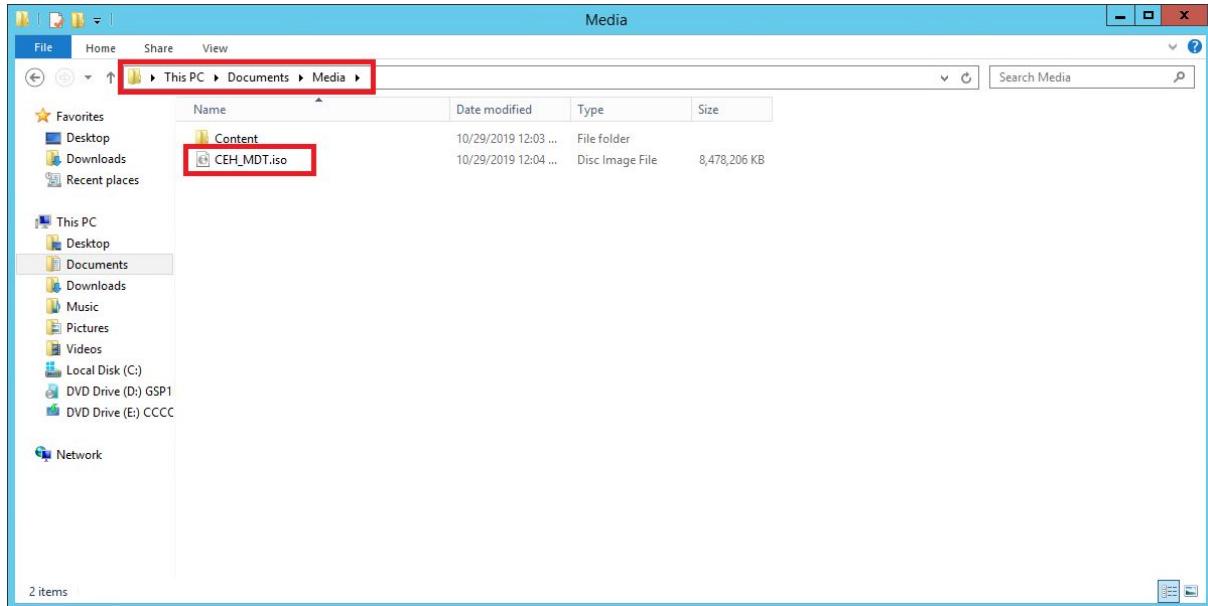
Update the media by right click on the media and then click "Update Media Content".



Wait for the media to be updated.

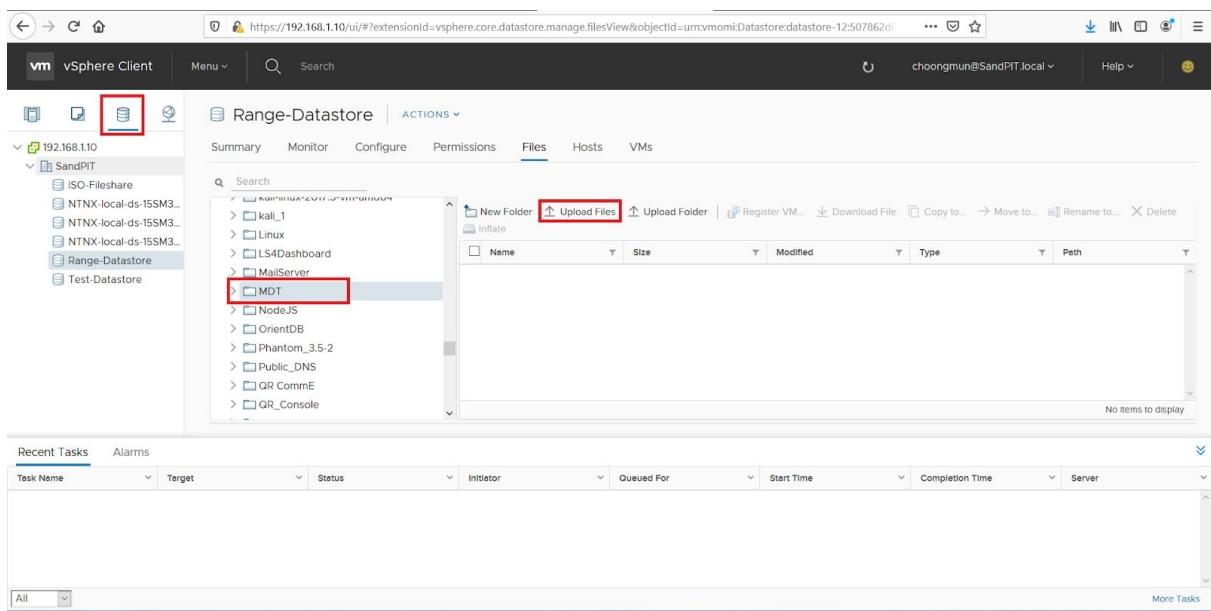


The ISO image can be found on the path you specified.

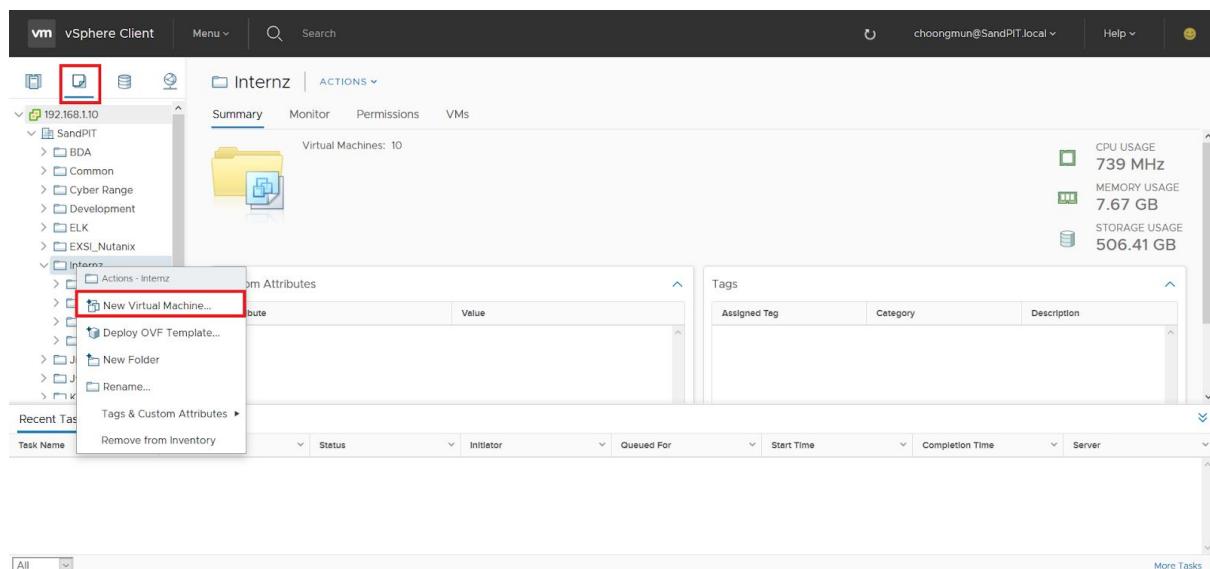


## Deploying ISO Image on Sandpit

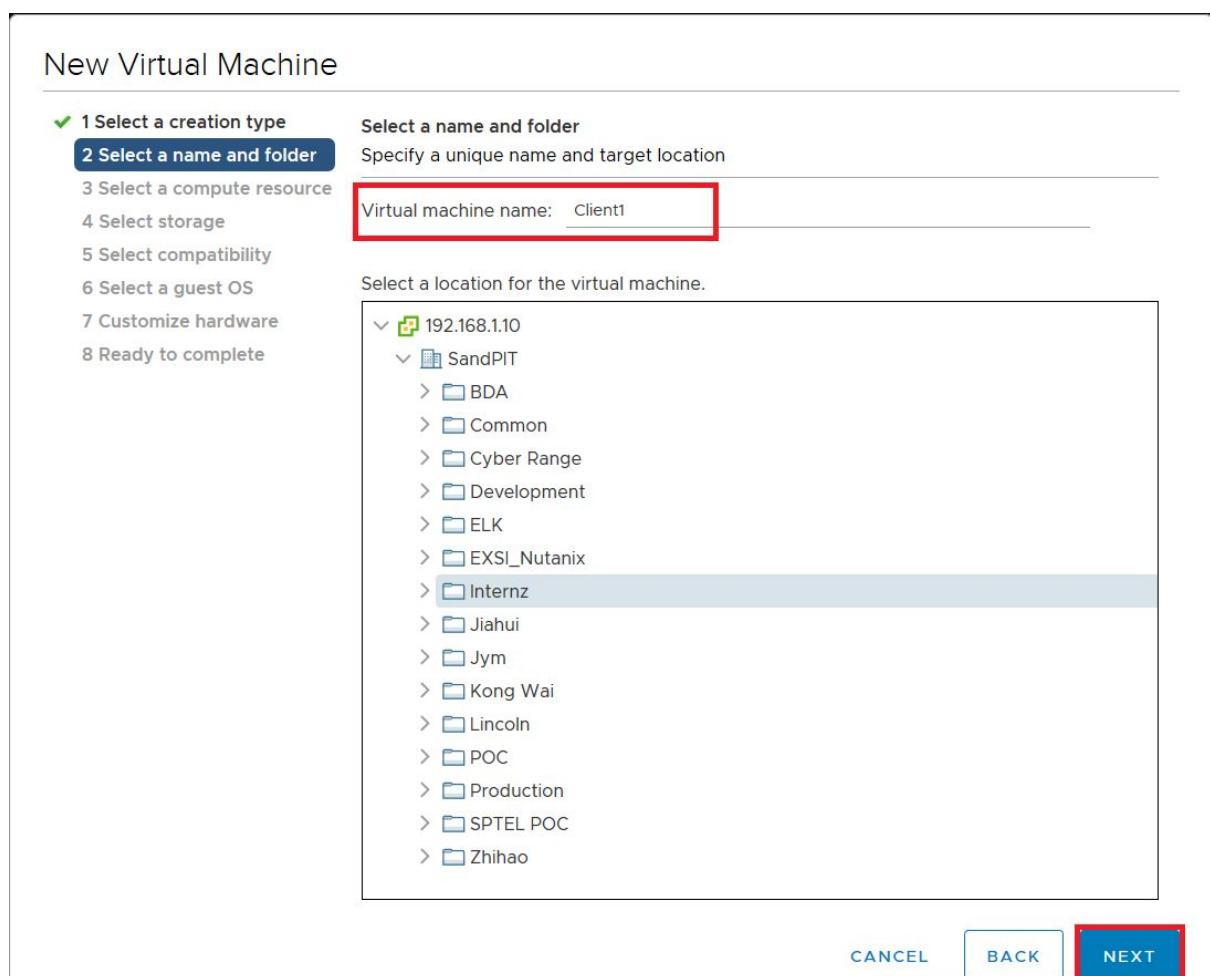
Upload the ISO image to sandpit. Click on “Storage”, create a folder and upload the ISO image by clicking on “Upload Files”.



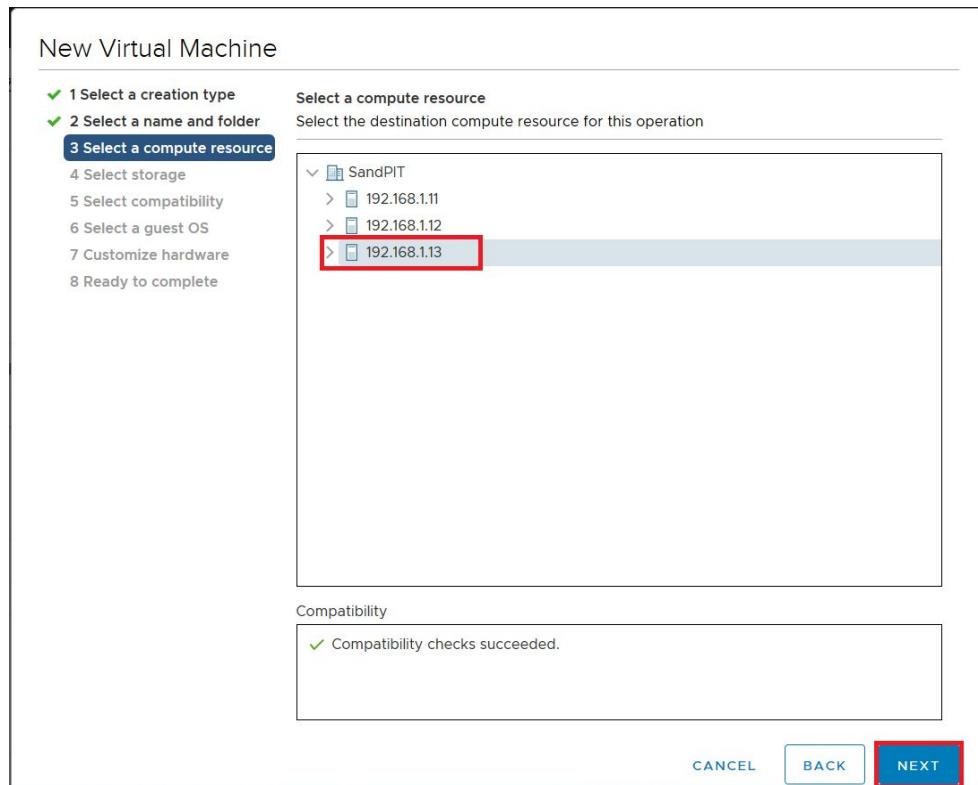
Click on “VMs and Templates” and right click on the folder that you to place your VM. Click on “New Virtual Machine” to create VM.



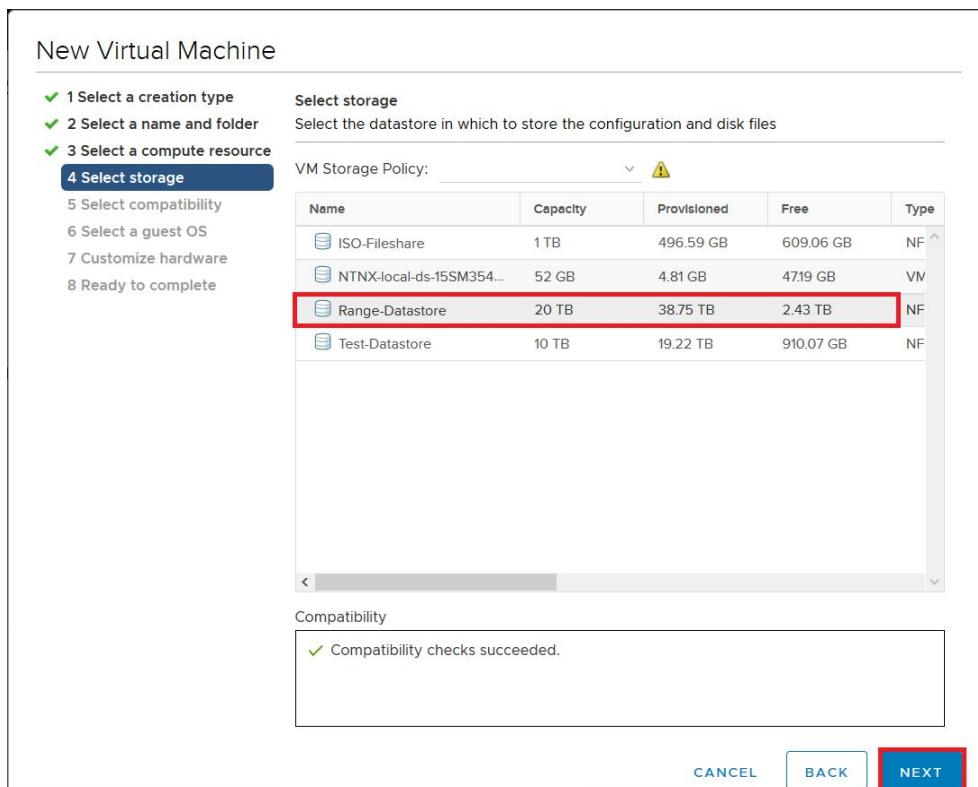
Click on “Next”, specify your VM’s name and then click “Next”.



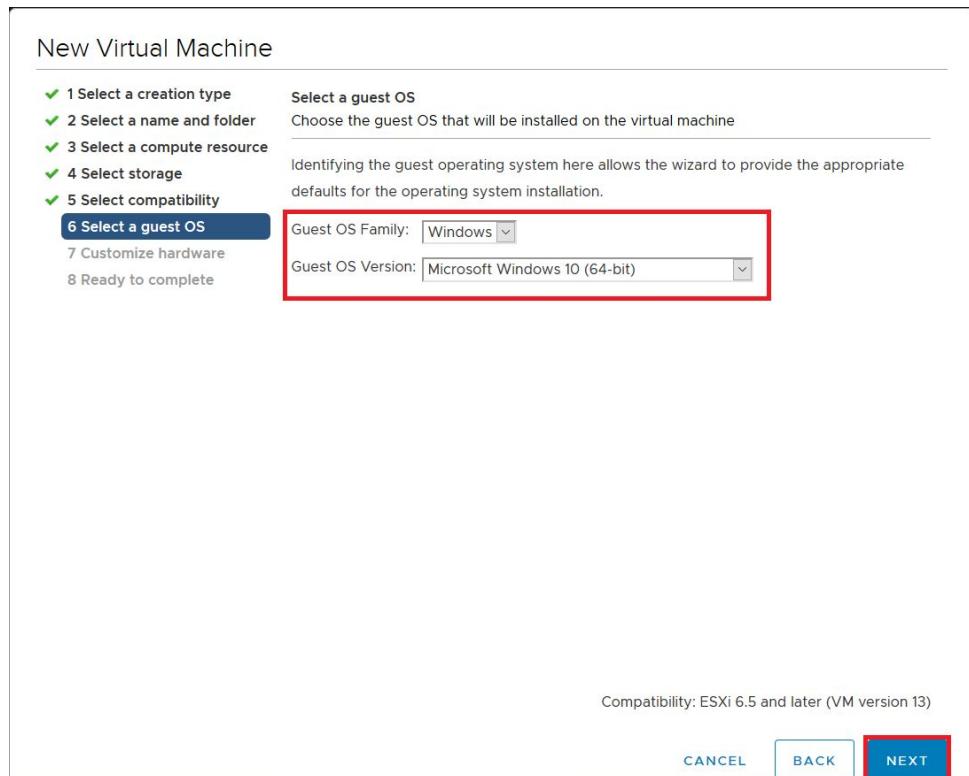
Specify which host to store the VM and click “Next”.



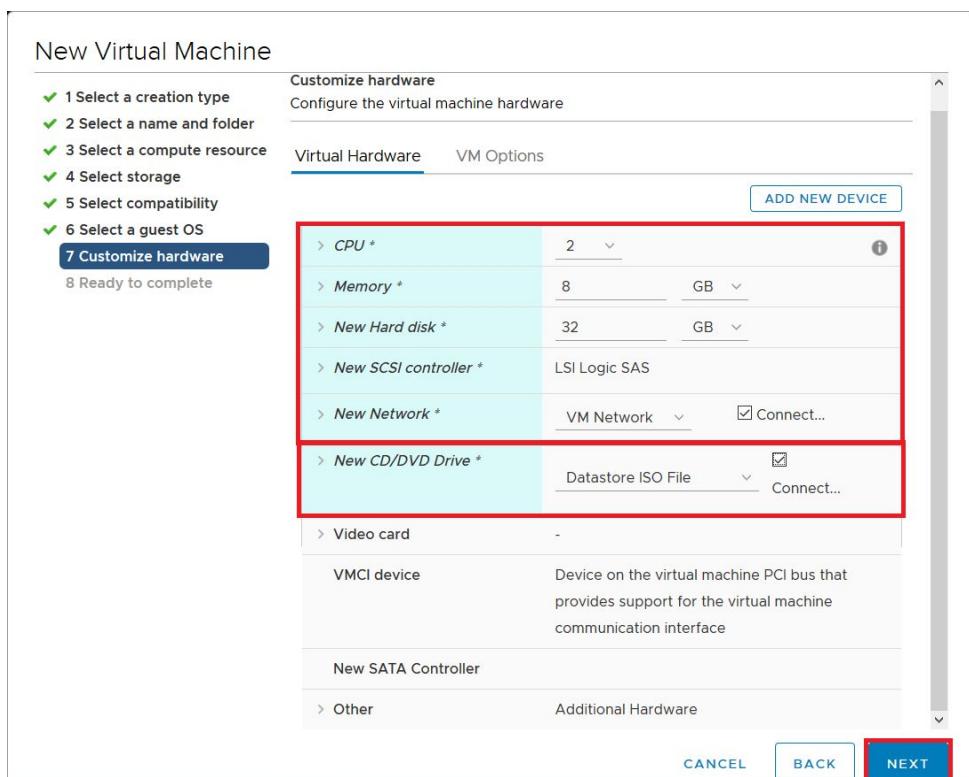
Specify which data storage to store the VM and click “Next”.



Click on “Next”, select your OS and click on “Next”.



Specify the VM's hardware specification, locate the ISO image on “New CD/DVD Drive”, click “Next” and click “Finish” on the summary page.

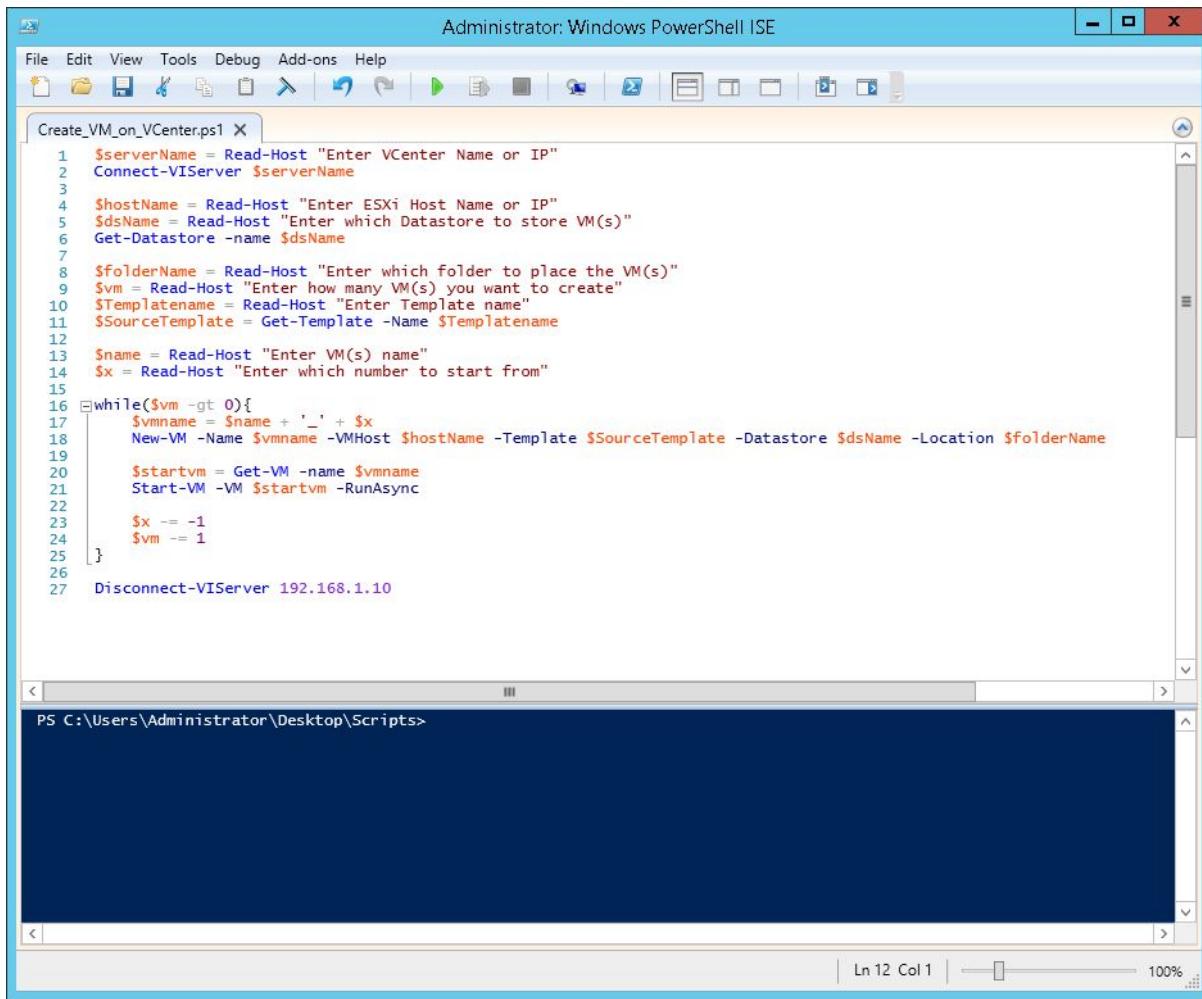


## Script to Deploy VM(s) on Sandpit

VMware PowerCLI will be used to deploy the VM(s) on sandpit. The commands for it can be found [here](#).

The script to first ask the user which server to connect to and then connect to the server with the command “Connect-VI-Server”. Then it will ask the user which ESXi host, Datastore and folder to store the VM(s). It will then ask the user which template to clone from. afterwhich, it will ask the user to specify the name of the VM(s) and how many to create.

The command “New-VM” is used to create VM and the command “Start-VM” is used to power on the VM.



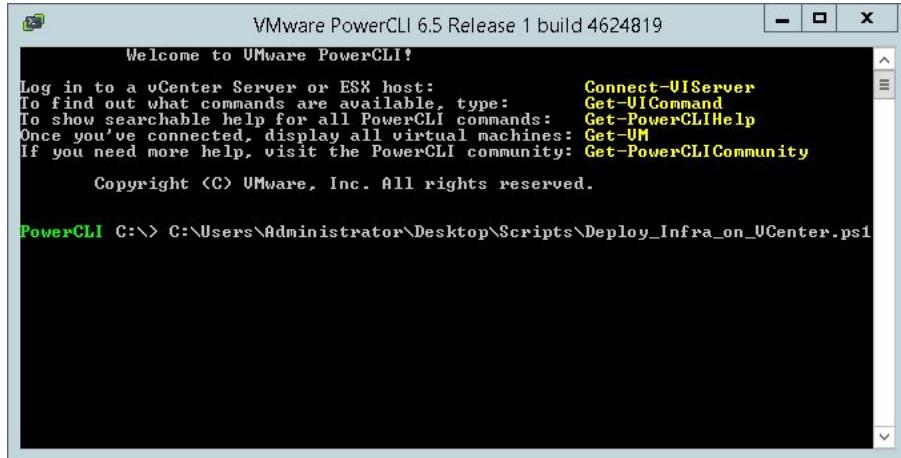
The screenshot shows the Windows PowerShell Integrated Scripting Environment (ISE) window titled "Administrator: Windows PowerShell ISE". The script file is named "Create\_VM\_on\_VCenter.ps1". The code in the script is as follows:

```
1 $serverName = Read-Host "Enter VCenter Name or IP"
2 Connect-VIServer $serverName
3
4 $hostName = Read-Host "Enter ESXi Host Name or IP"
5 $dsName = Read-Host "Enter which Datastore to store VM(s)"
6 Get-Datastore -name $dsName
7
8 $folderName = Read-Host "Enter which folder to place the VM(s)"
9 $vm = Read-Host "Enter how many VM(s) you want to create"
10 $templatename = Read-Host "Enter Template name"
11 $sourceTemplate = Get-Template -Name $templatename
12
13 $name = Read-Host "Enter VM(s) name"
14 $x = Read-Host "Enter which number to start from"
15
16 while($vm -gt 0){
17     $vmname = $name + '_' + $x
18     New-VM -Name $vmname -VMHost $hostName -Template $sourceTemplate -Datastore $dsName -Location $folderName
19
20     $startvm = Get-VM -name $vmname
21     Start-VM -VM $startvm -RunAsync
22
23     $x -= -1
24     $vm -= 1
25 }
26
27 Disconnect-VIServer 192.168.1.10
```

The bottom pane of the ISE shows a PowerShell session window with the command "PS C:\Users\Administrator\Desktop\Scripts>" followed by a blank dark blue area. The status bar at the bottom right indicates "Ln 12 Col 1" and "100%".

## Deploying VM(s) on Sandpit

To deploy VM(s), just drag and drop the script into PowerCLI and run it.



Welcome to VMware PowerCLI!

Log in to a vCenter Server or ESX host: Connect-UIServer  
To find out what commands are available, type: Get-UICommand  
To show searchable help for all PowerCLI commands: Get-PowerCLIHelp  
Once you've connected, display all virtual machines: Get-VM  
If you need more help, visit the PowerCLI community: Get-PowerCLICommunity

Copyright <C> VMware, Inc. All rights reserved.

PowerCLI C:\> C:\Users\Administrator\Desktop\Scripts\Deploy\_Infra\_on\_UCenter.ps1

The script will start to run.



00DA640920E542B33E

Clone virtual machine  
percent complete: 36  
[ooooooooooooooooooooooo]

[Thumbprint]  
3AE7235836FF6B708A3F8C48A6E801B6071A8693

The server certificate is not valid.  
WARNING: THE DEFAULT BEHAVIOR UPON INVALID SERVER CERTIFICATE WILL CHANGE IN A FUTURE RELEASE. To ensure scripts are not affected by the change, use Set-PowerCLIConfiguration to set a value for the InvalidCertificateAction option.

Name	Port	User
192.168.1.10	443	SANDPIT.LOCAL\choongmun