

IT2654 System Administration and Security
Practical 6B - Installing and Configuring BitLocker
and AppLocker

Overview

To configure and administer Windows BitLocker and AppLocker

Objectives

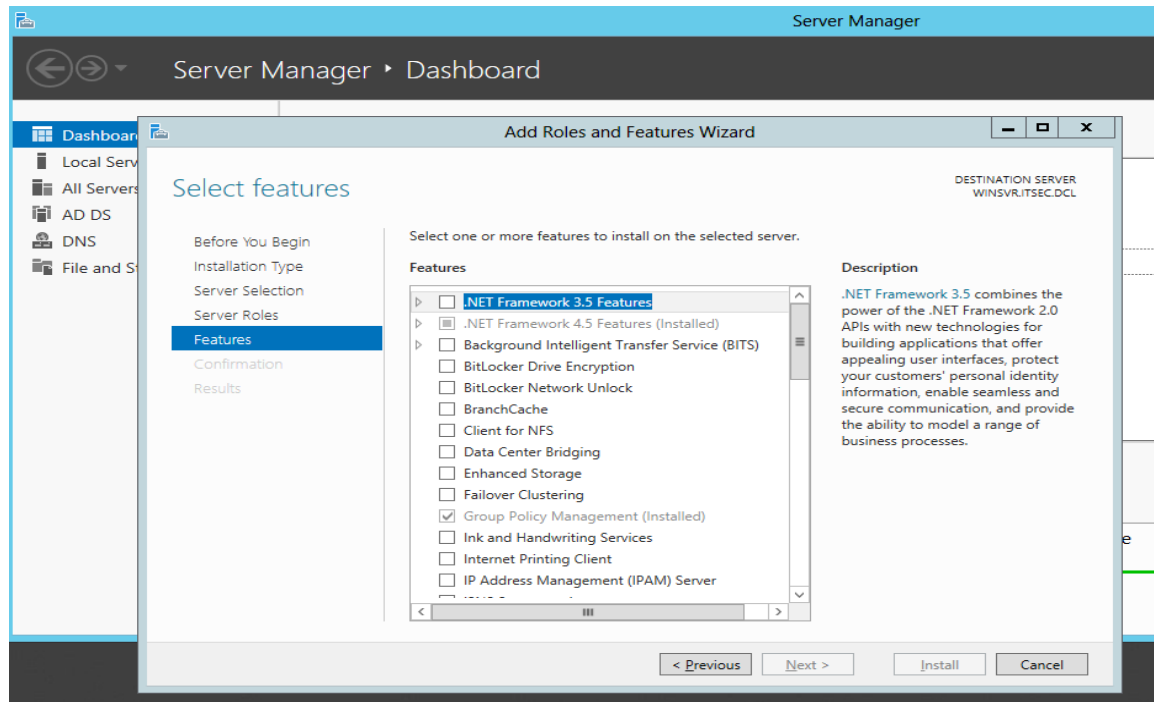
Task 1 – Security Administration using Windows BitLocker

Lab Requirements

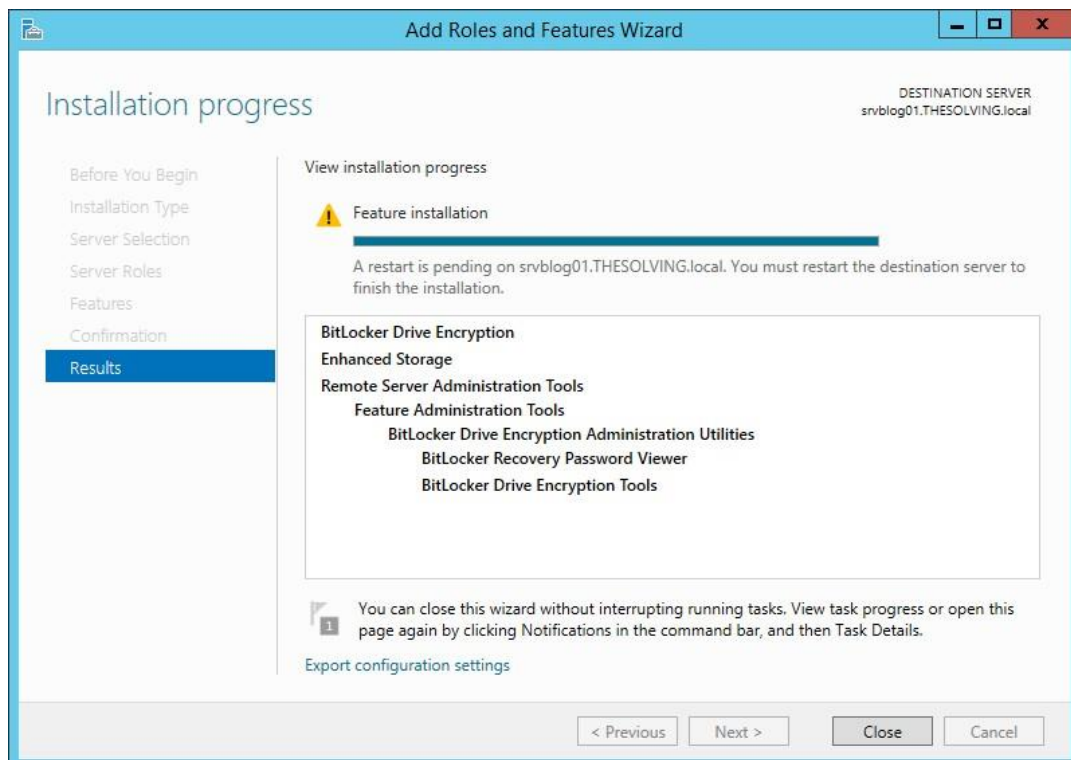
- Windows Server 2016 NYP-DC1

Microsoft allows to encrypt the disks of a server with a feature named **BitLocker**. We are going to see how you can enable **BitLocker** on a virtual server to protect your company from data theft for this lab.

1. Before starting, take a snapshot of your Windows Server NYP-DC1 VM (Virtual Machine). Go to **VM → Snapshot → Take Snapshot**. Enter the name “**b4bitLocker**” and click **Take Snapshot**. This will create a copy of your VM. You can revert to this copy should anything happen to your virtual machine.
2. Login to NYP-DC1 as Administrator. BitLocker is **a feature NOT a role**. Open **Server Manager**. We will Install the **BitLocker Drive Encryption** feature with the *Add Roles and Features Wizard*:



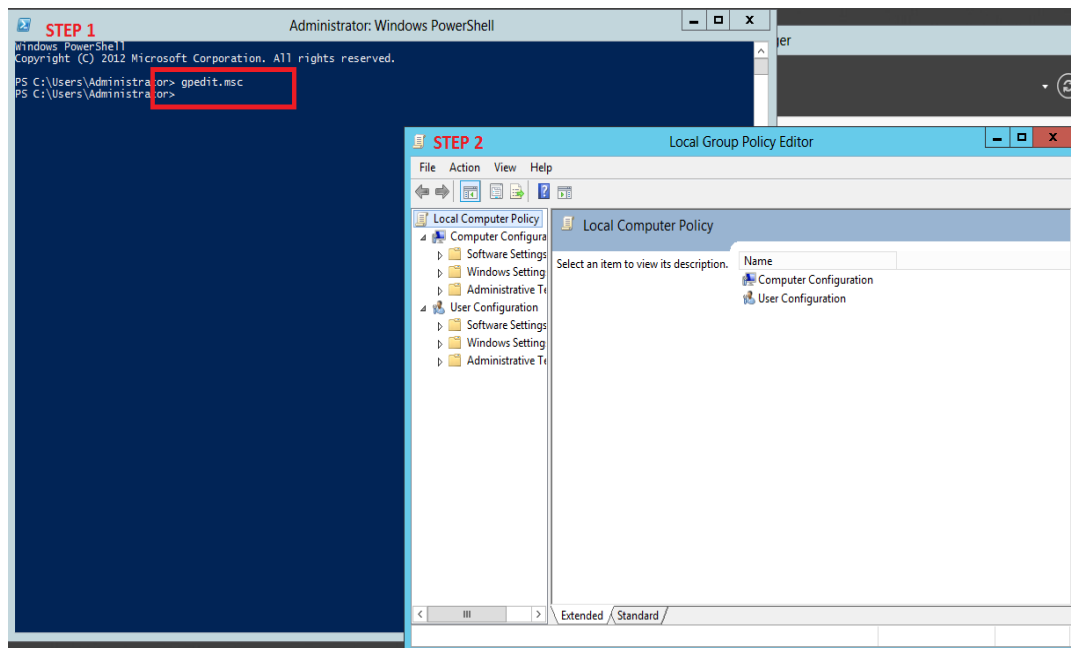
- Follow the wizard to add the feature. You need to restart the system after the installation:



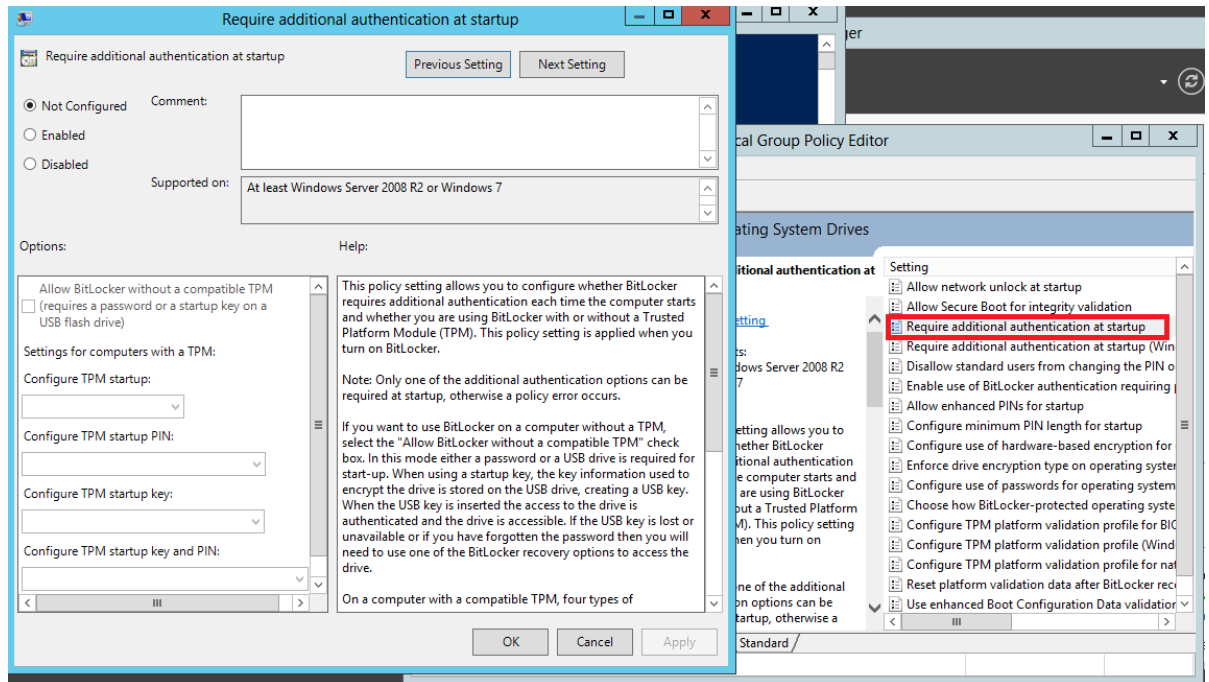
Restart the server. You can do that by CTRL-ALT-INSERT within the

4. You need the **Trusted Platform Module (TPM)** in order to take advantage of **BitLocker** encryption. Virtual machines do not have the **TPM module** so you need to follow these two steps **BEFORE** configuring **BitLocker** (**BitLocker** must be installed on the server).

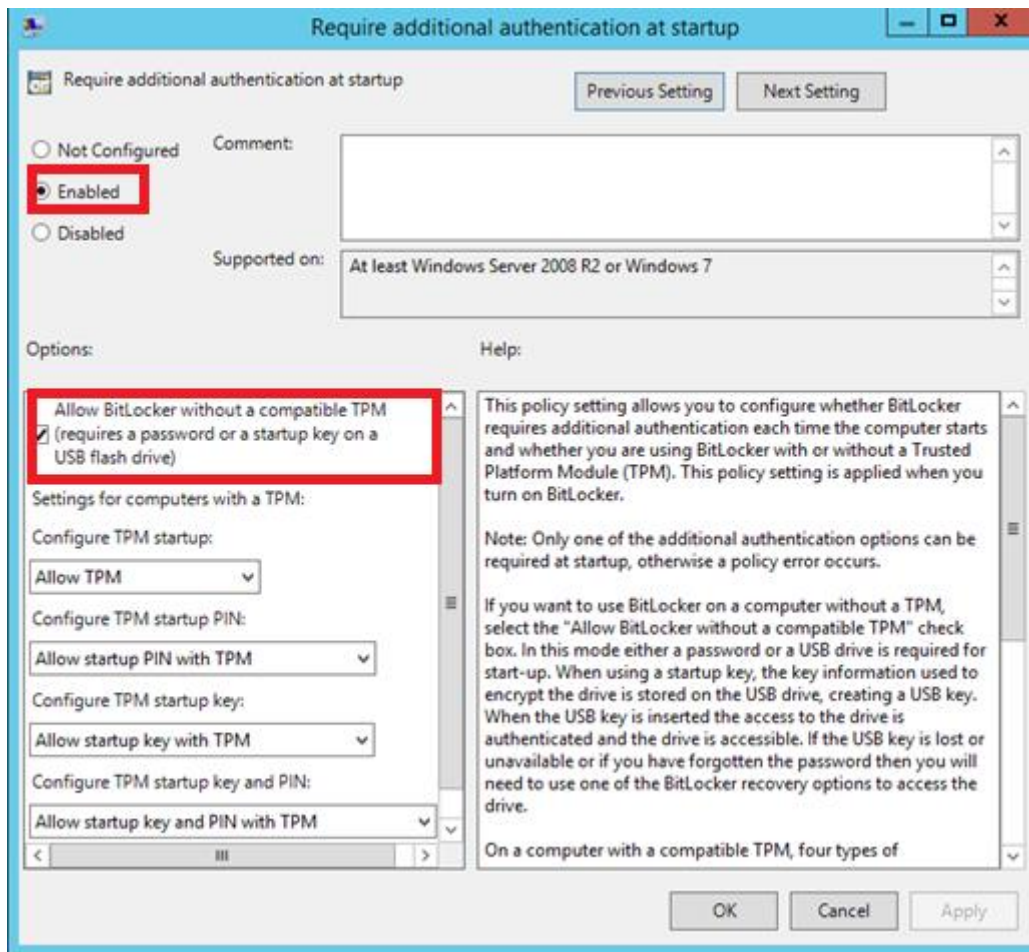
Click on the Search icon (magnifying glass) at the bottom left of the Windows Server screen. Type **gpedit.msc** to open the *Local Group Policy Editor*.



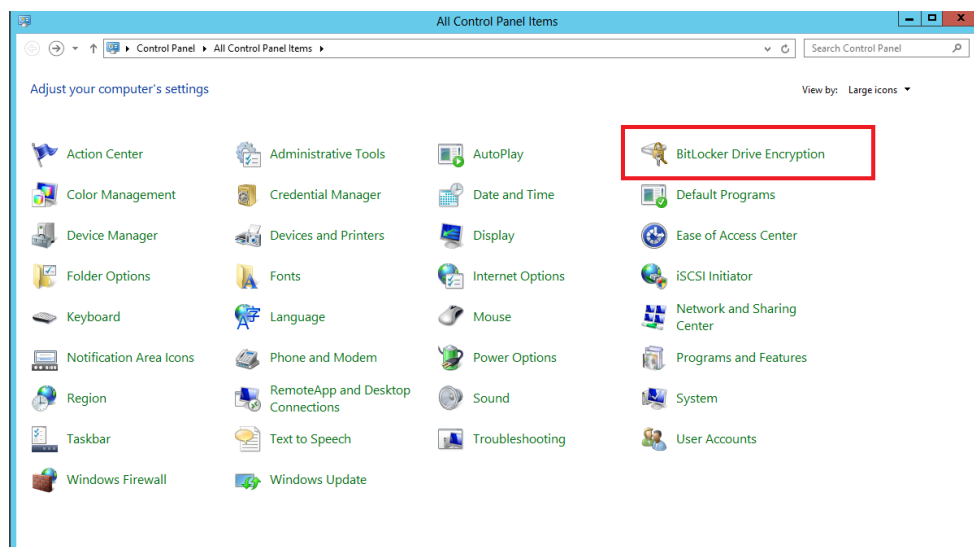
5. Go to Computer Configuration->Administrative Templates->Windows Components->BitLocker Drive Encryption->Operating System Drives. Double-click **Require additional authentication at startup**:

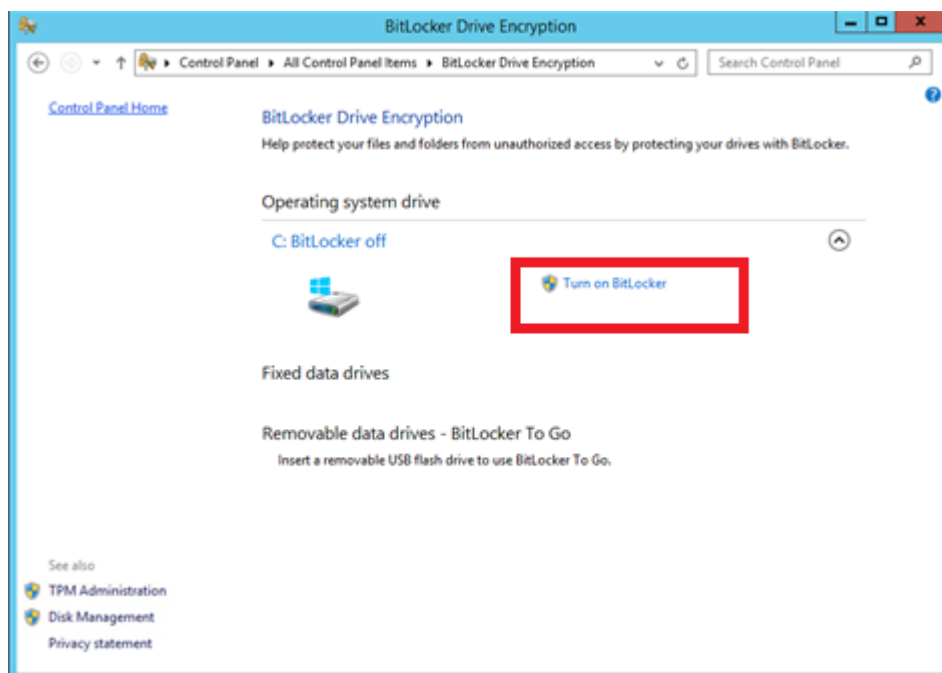


6. Select *Enable* and check *Allow BitLocker without a compatible TPM*:

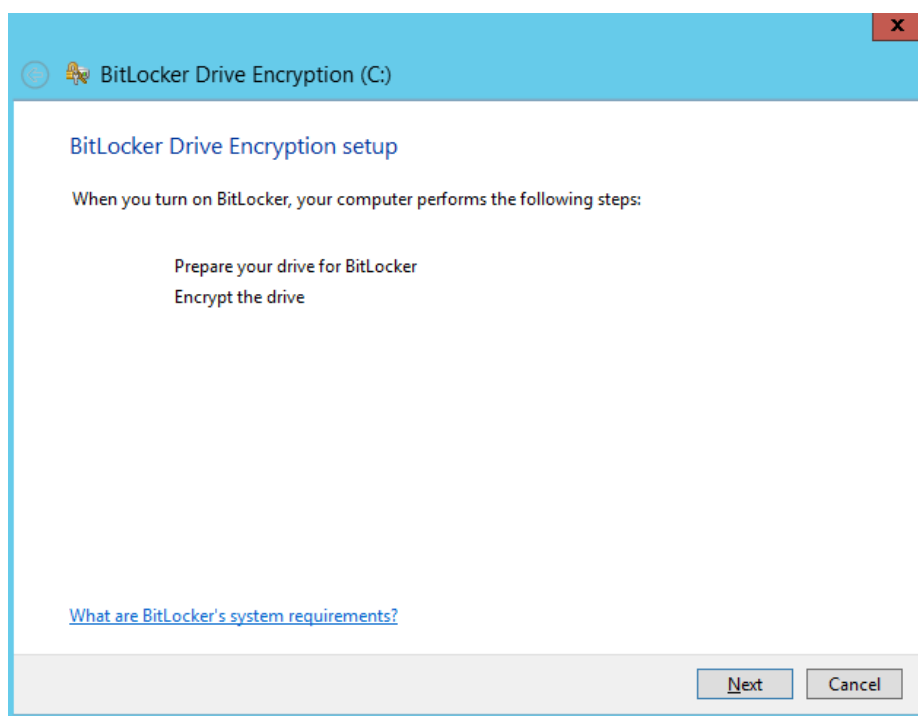


7. Restart the server. Open *Control Panel*, you'll find the *BitLocker* configuration panel. Open it and click *Turn On BitLocker*:

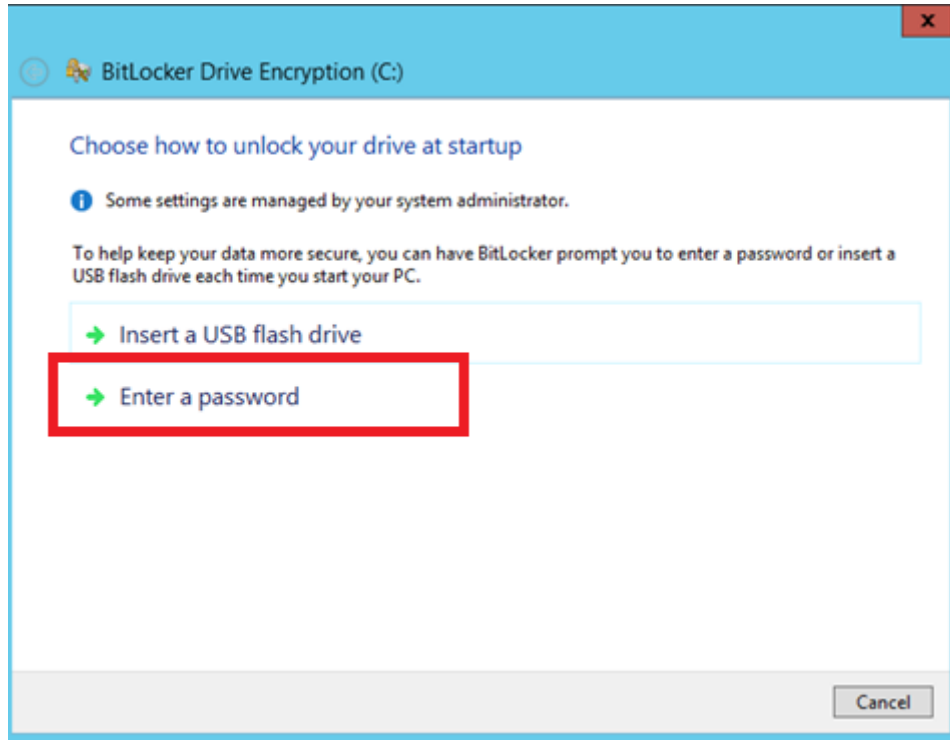




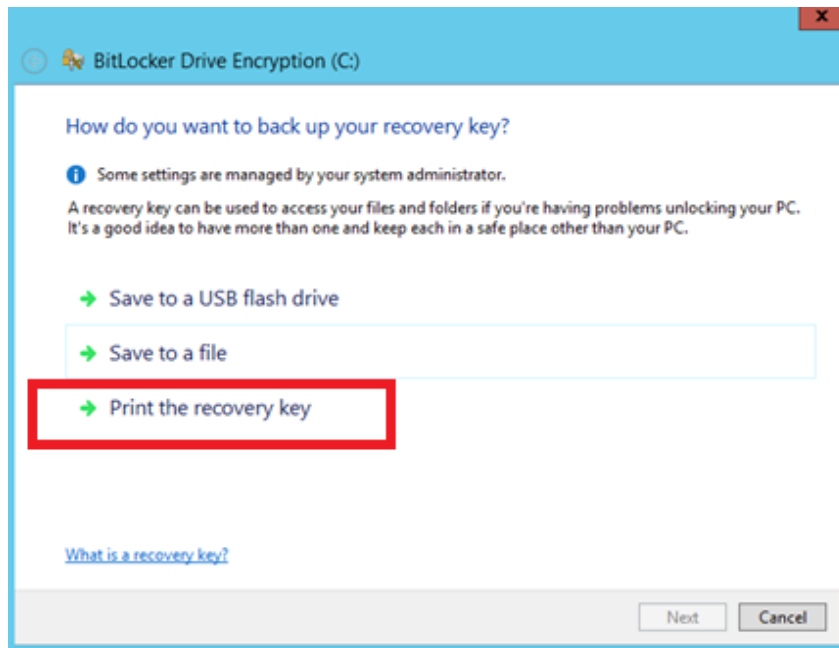
8. Click Next to start the process



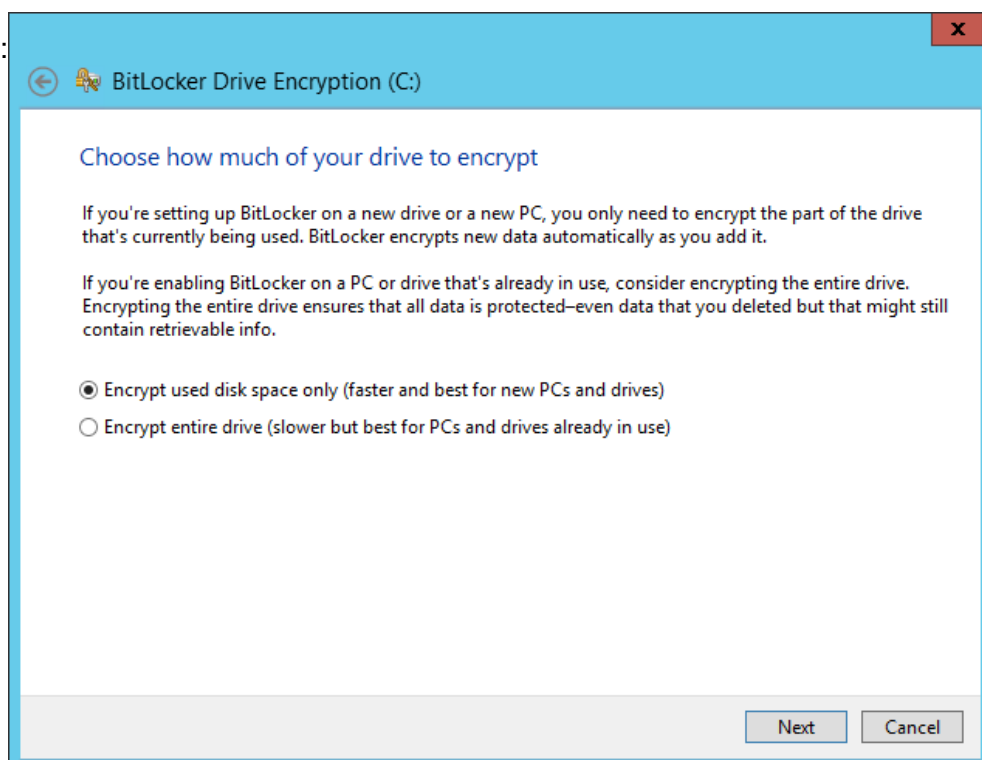
9. In this lab, we are using a **Virtual Machine**, so a system without a *TPM*, and *Windows will* ask us to configure an additional authentication at startup. We **chose a password to protect the data (use standard password Pa\$\$w0rd)**.





10. A recovery key can save you from big troubles. We will print it for security reasons: And save it in Microsoft XPS format. Give the file a filename of your choice.




11. Choose the encryption option "Encrypt used disk space only (faster and best for new PCs and drives)"



12. Choose which encryption mode to use – click Next to select default option.

  BitLocker Drive Encryption (C:)



Choose which encryption mode to use

Windows 10 (Version 1511) introduces a new disk encryption mode (XTS-AES). This mode provides additional integrity support, but it is not compatible with older versions of Windows.

If this is a removable drive that you're going to use on older version of Windows, you should choose Compatible mode.

If this is a fixed drive or if this drive will only be used on devices running at least Windows 10 (Version 1511) or later, you should choose the new encryption mode

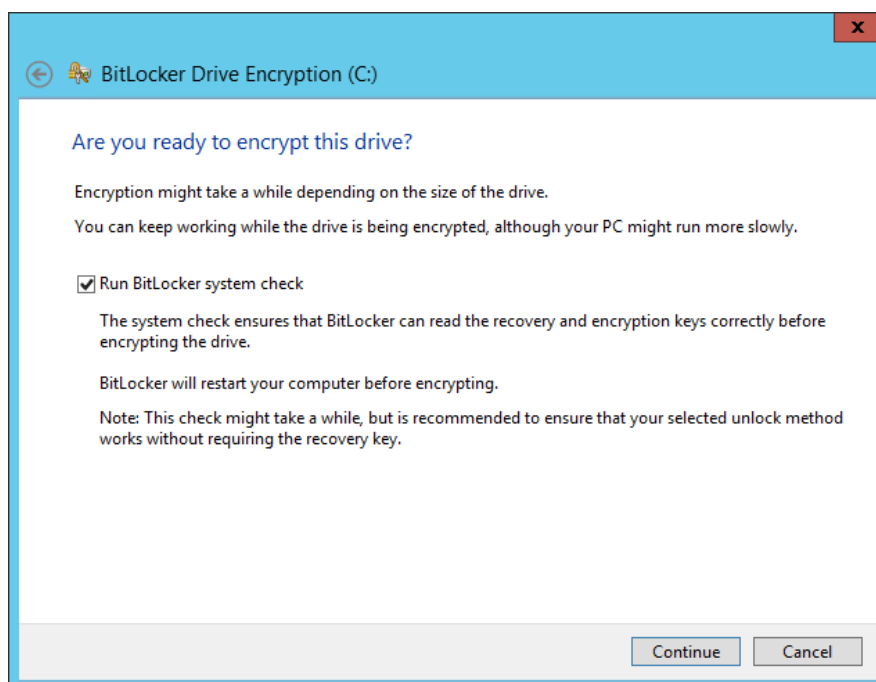
☒ New encryption mode (best for fixed drives on this device)

☐ Compatible mode (best for drives that can be moved from this device)

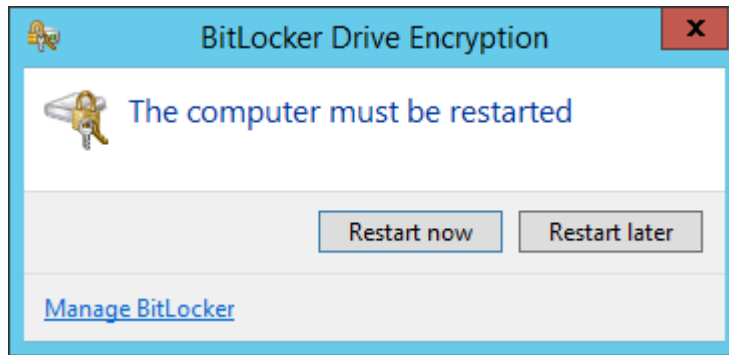
Next

Cancel

13. Click *Continue*:



13. Restart the system if it doesn't automatically restart.



14. At the next boot you'll be "forced" to enter the *password* or plug the *USB flash drive*. After the Windows start **BitLocker** will begin the encryption process:



Objectives

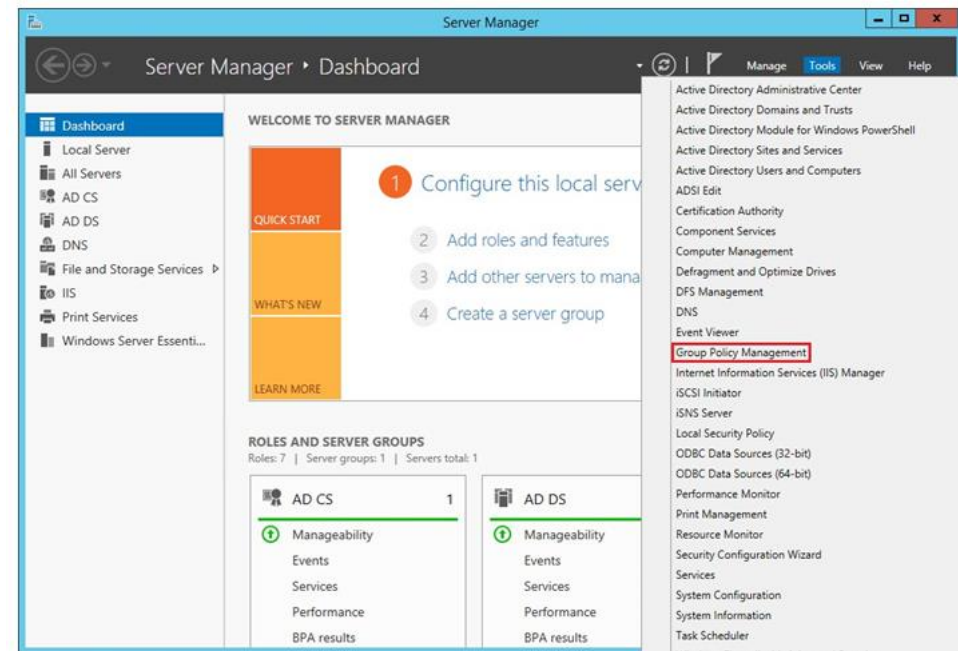
Task 2 – Security Administration using Windows AppLocker

In **Windows Server 2016**, **AppLocker** is a feature and a subset of **GPOs** to enforce software restriction.

AppLocker can manage execution permissions of:

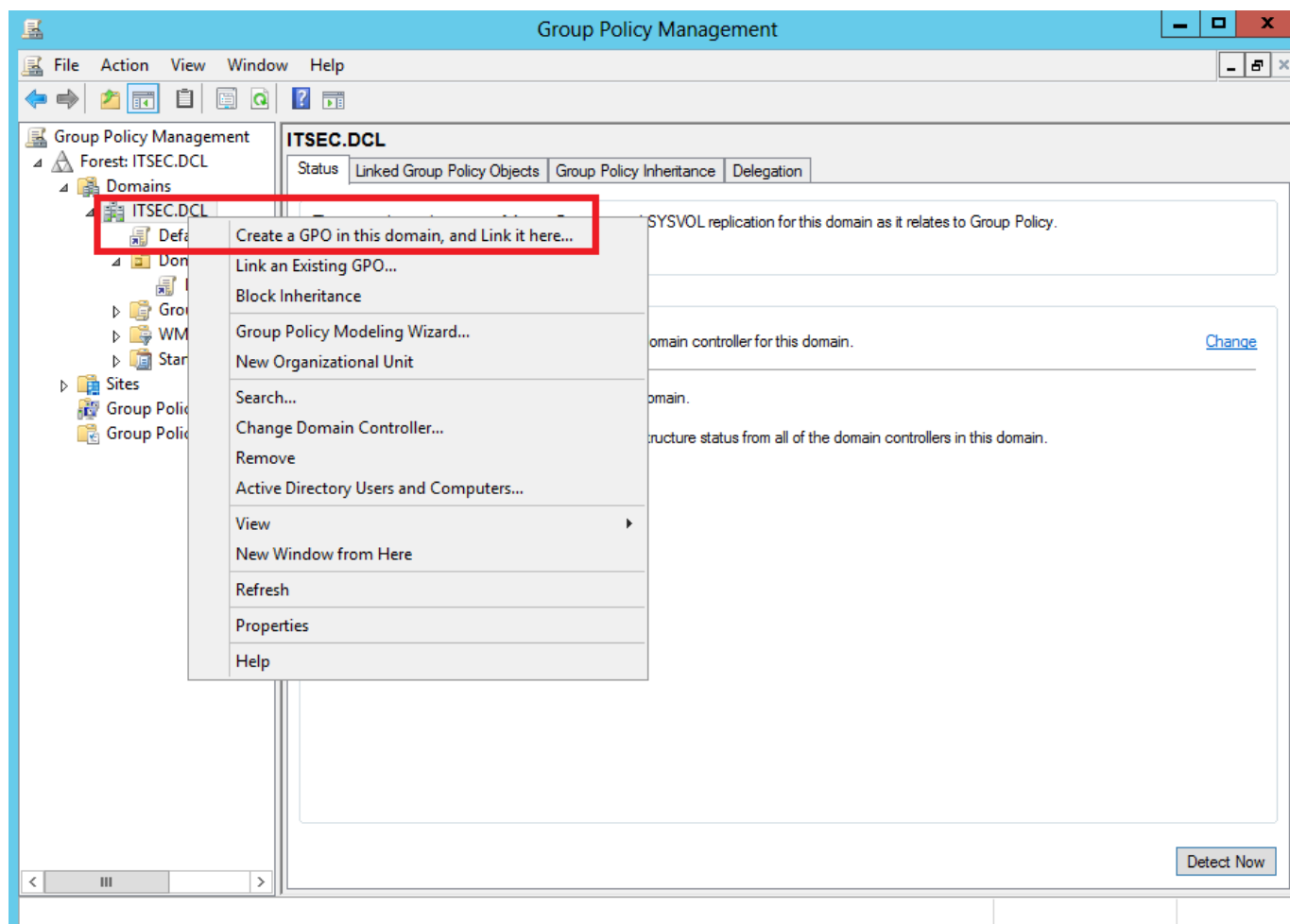
- **Executables:** files with .exe extension
- **Windows installers:** Windows installer packages with .msi and .msp extensions
- **Scripts:** files with .ps1, .bat, .cmd, .cbs and .js extensions
- **Packaged Apps:** Windows Store apps

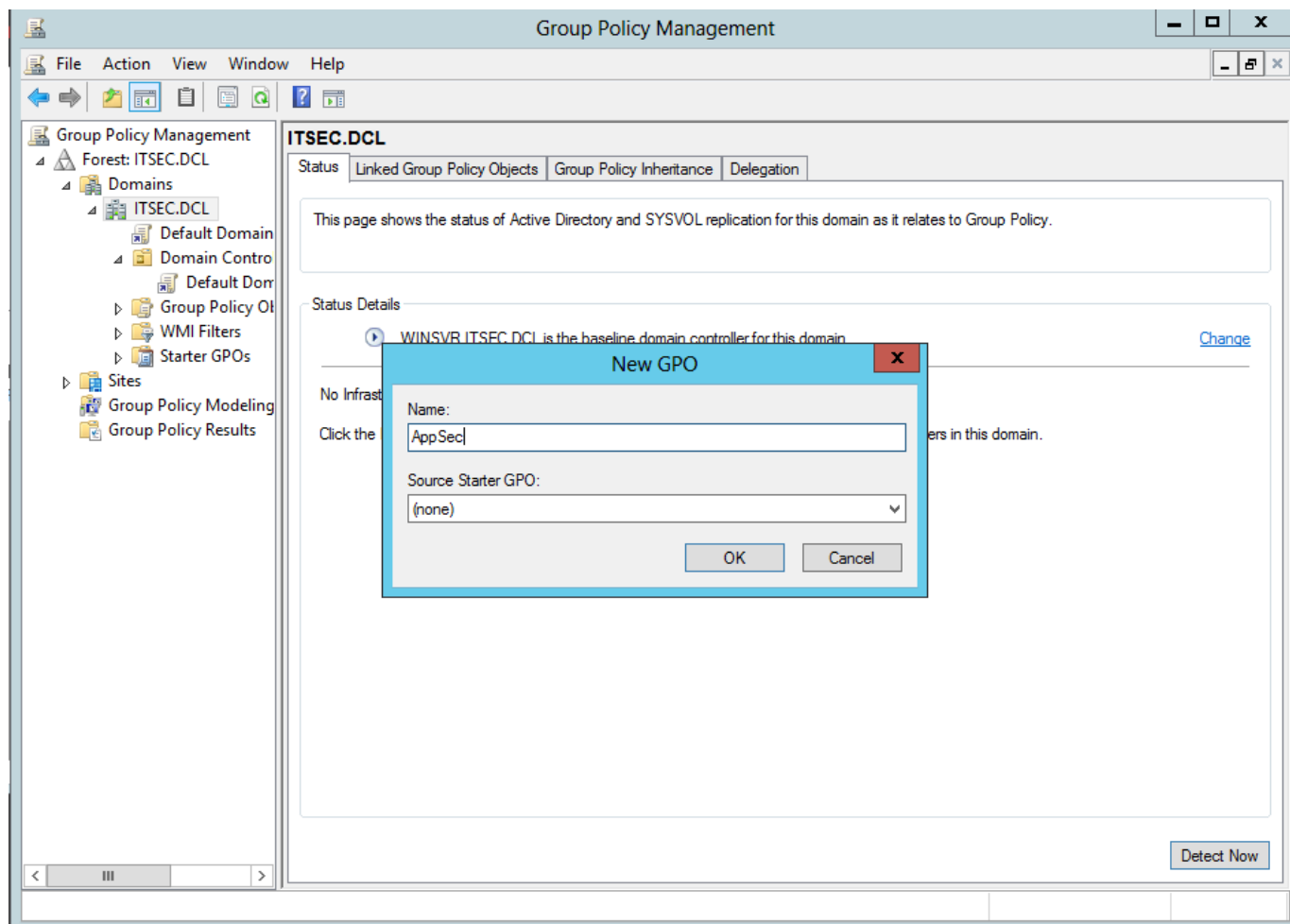
1. Login to NYP-DC1 as Administrator. Open the *Server Manager* and launch the *Group Policy Management*:



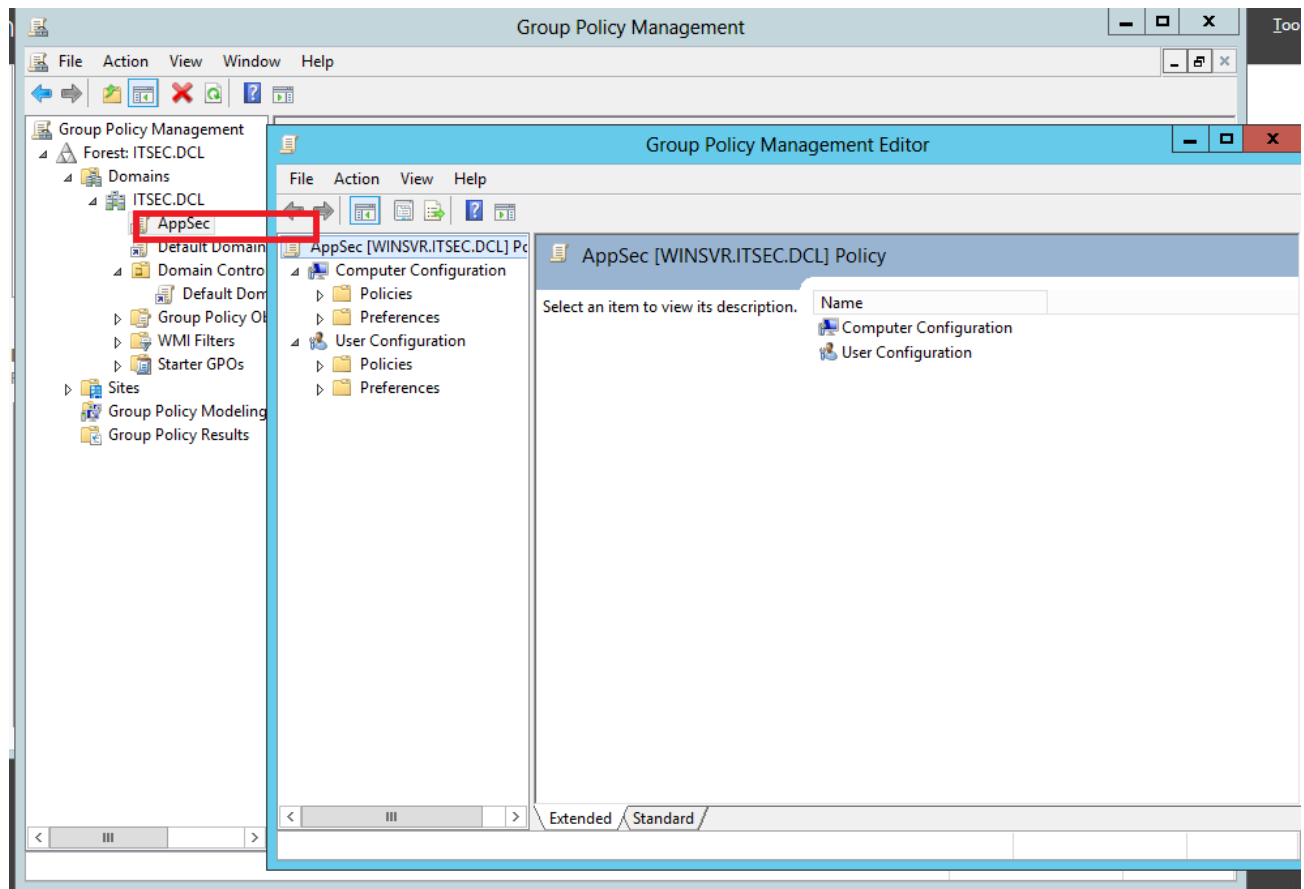
2. **Right-click "Frontier.net" domain** as shown and select "Create a GPO in this domain: " and give the policy a name **AppSec**.

(note: in the following screen shots, ITSEC.DCL is replaced by Frontier.net)

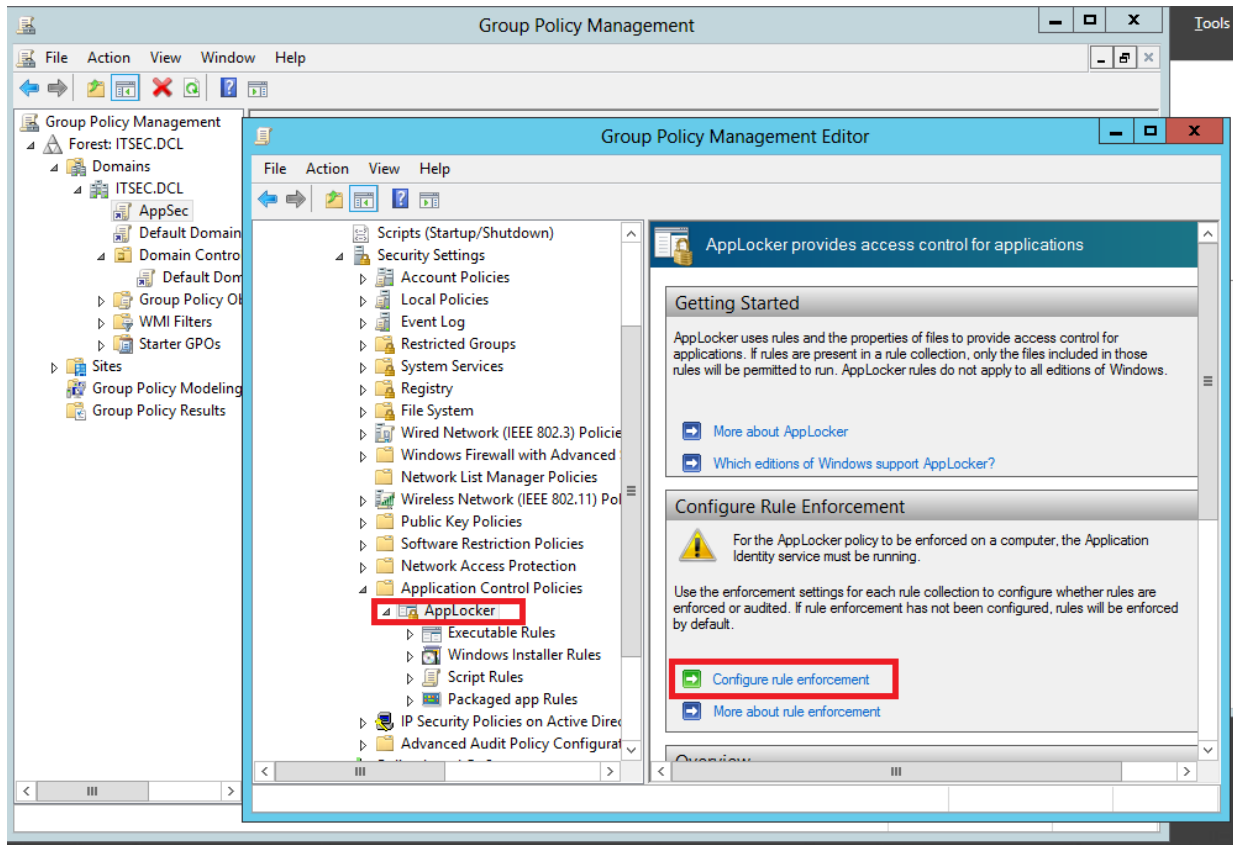




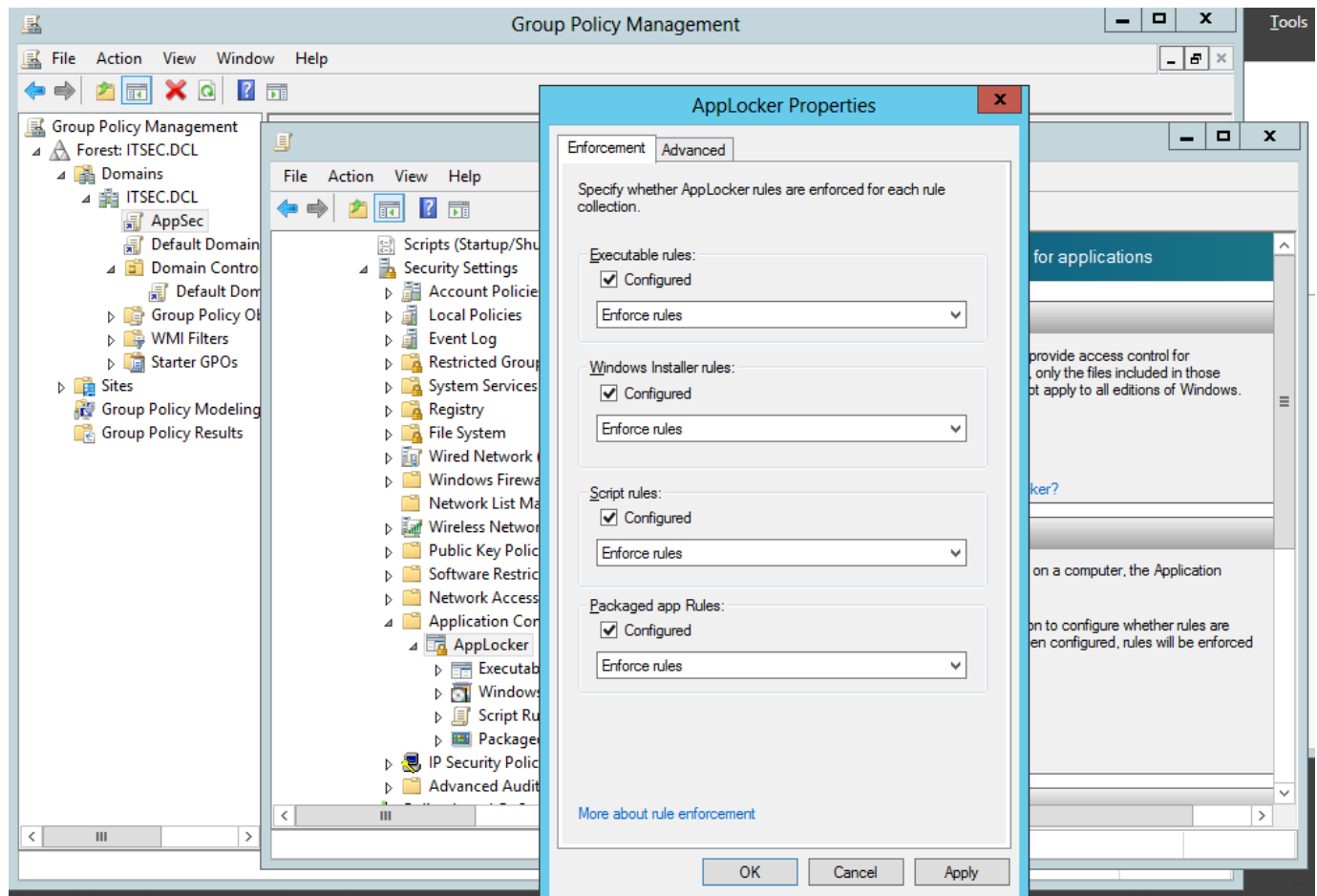
3. Right click on **AppSec** and Edit the policy:



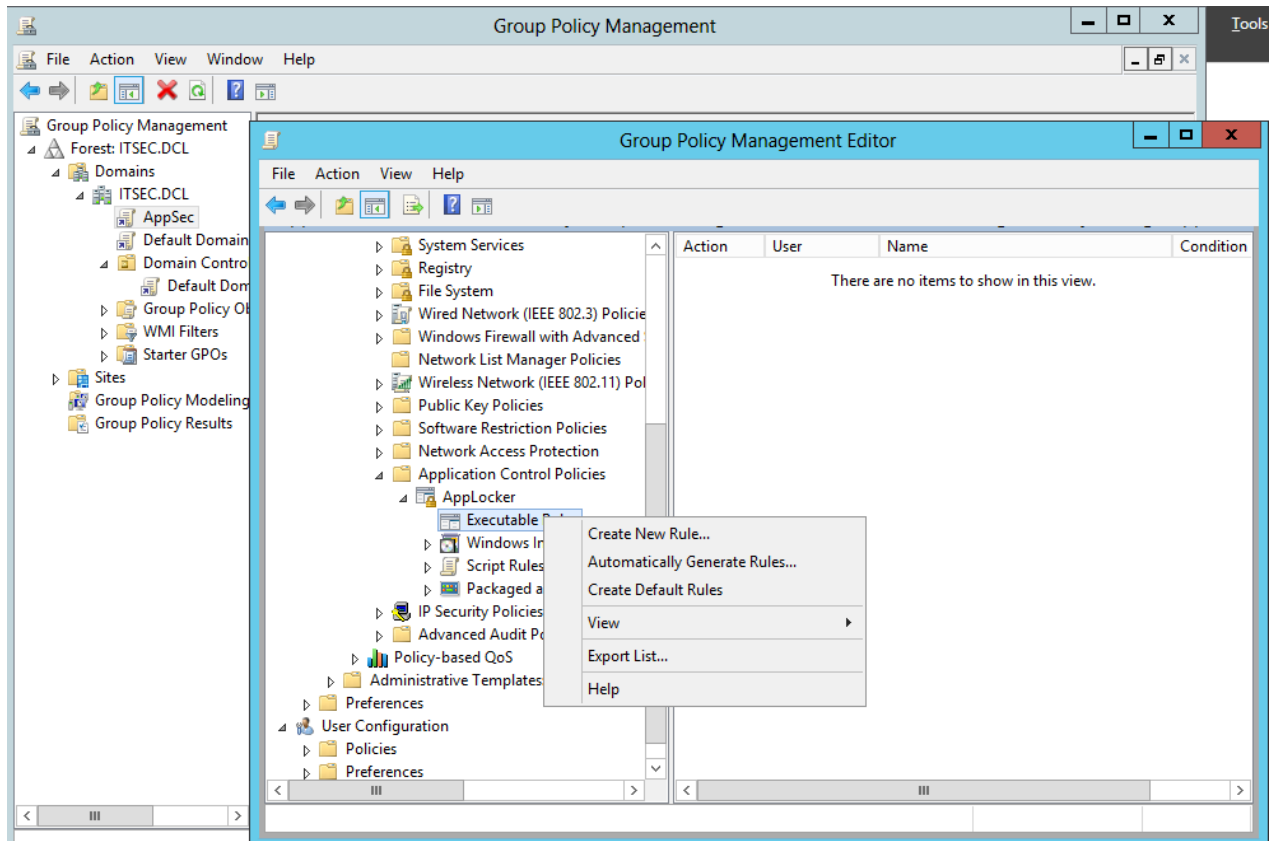
4. You will find the **AppLocker** settings inside the path **Computer Configuration->Policies->Windows Settings->Security Settings->Application Control Policies->AppLocker**. Click **Configure rule enforcement**:



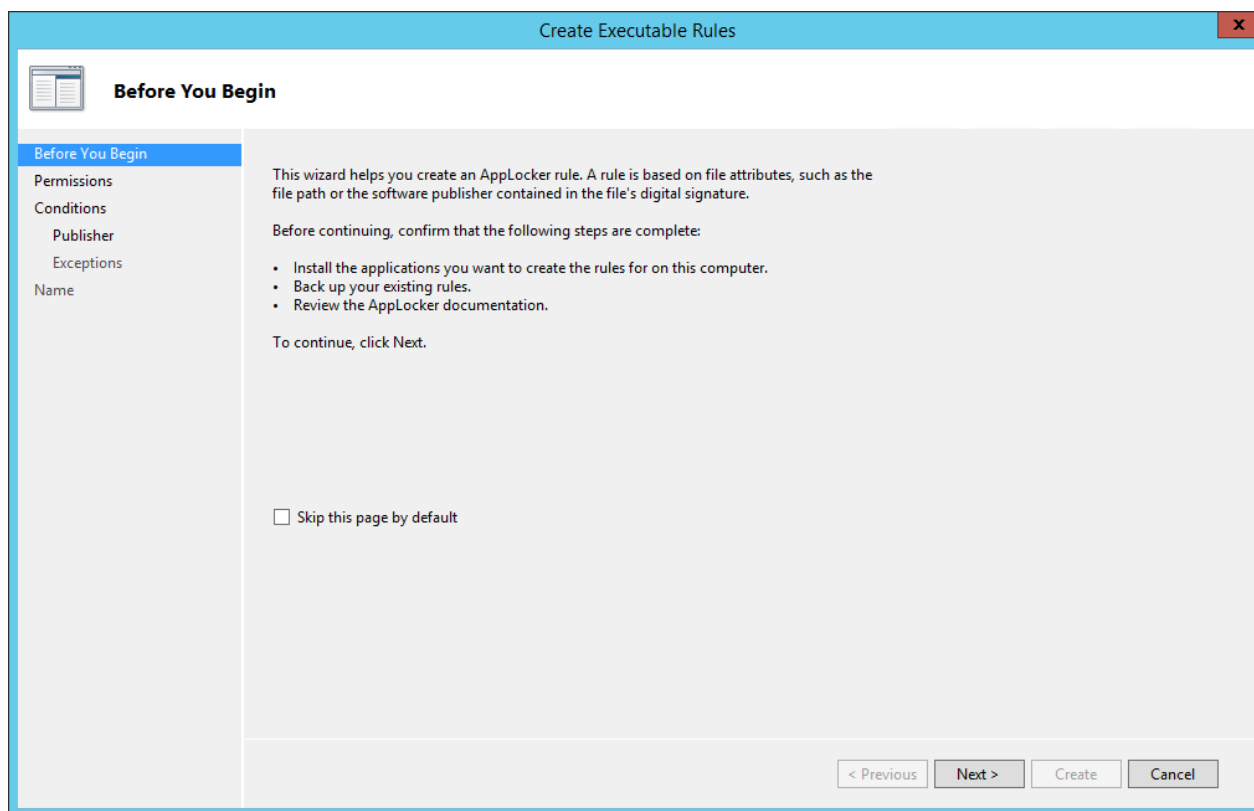
5. Configure the rules as shown below.:



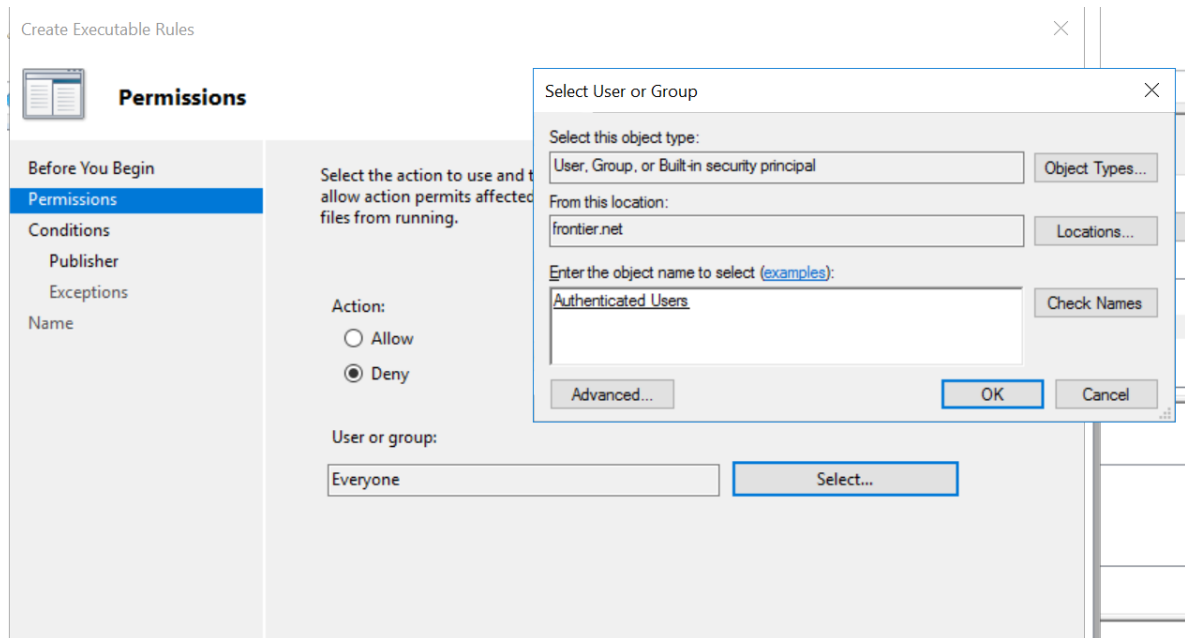
6. Under AppLocker, select **Executable Rule**. Right-click and choose **Create New Rule**:



7. Click *Next*:

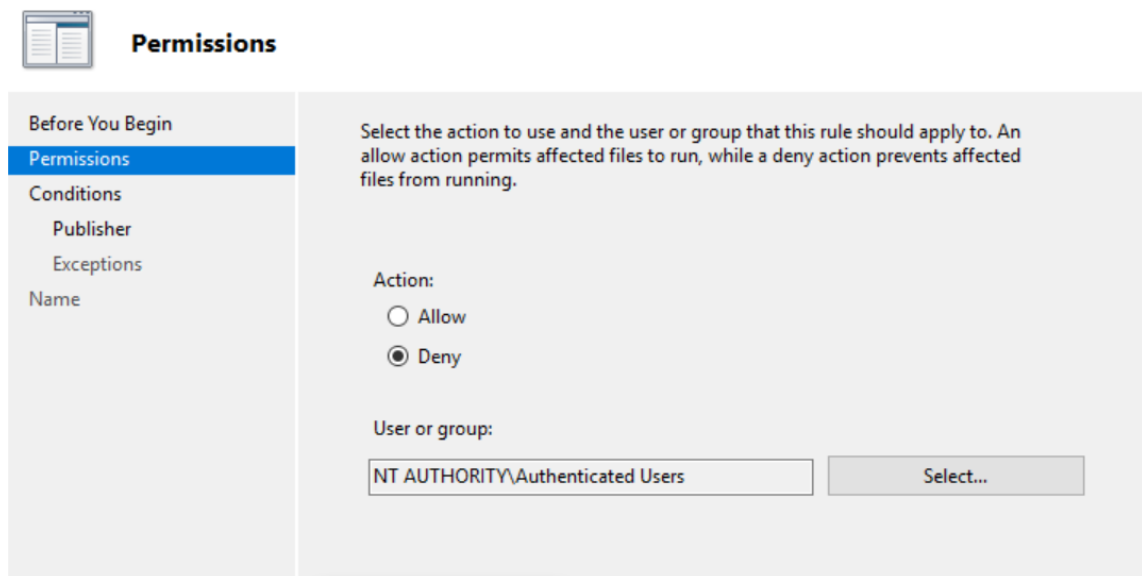


8. Specify the users who will be affected by the rule and the rule type (**Deny execution**). For your virtual machine, select **Authenticated Users**.



9. Then Click **Next**.

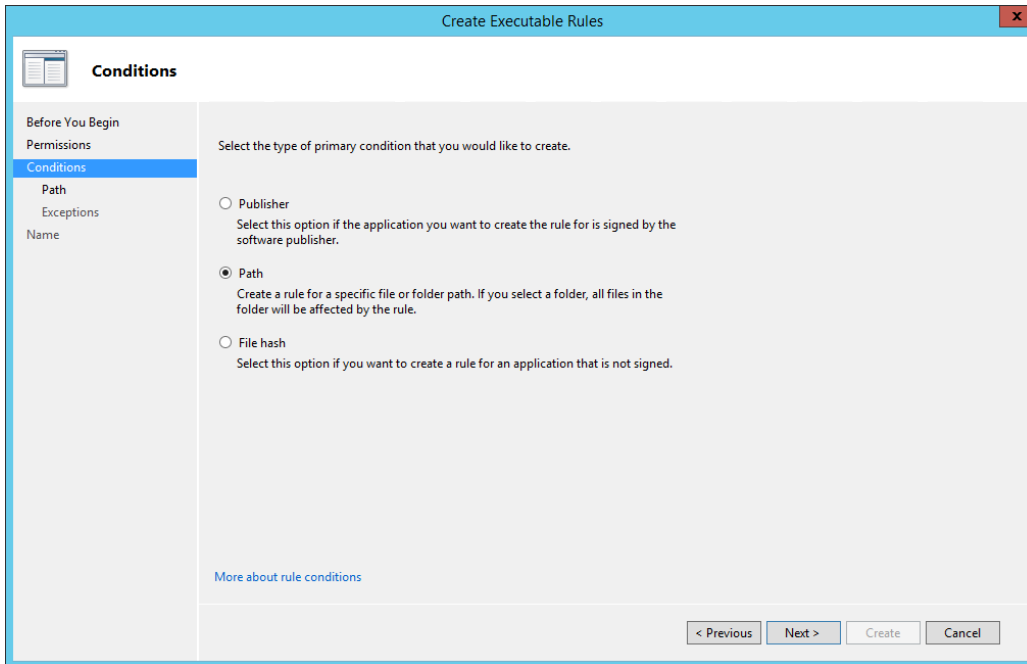
Create Executable Rules



10. There are three ways to specify which applications will be affected by the rule:

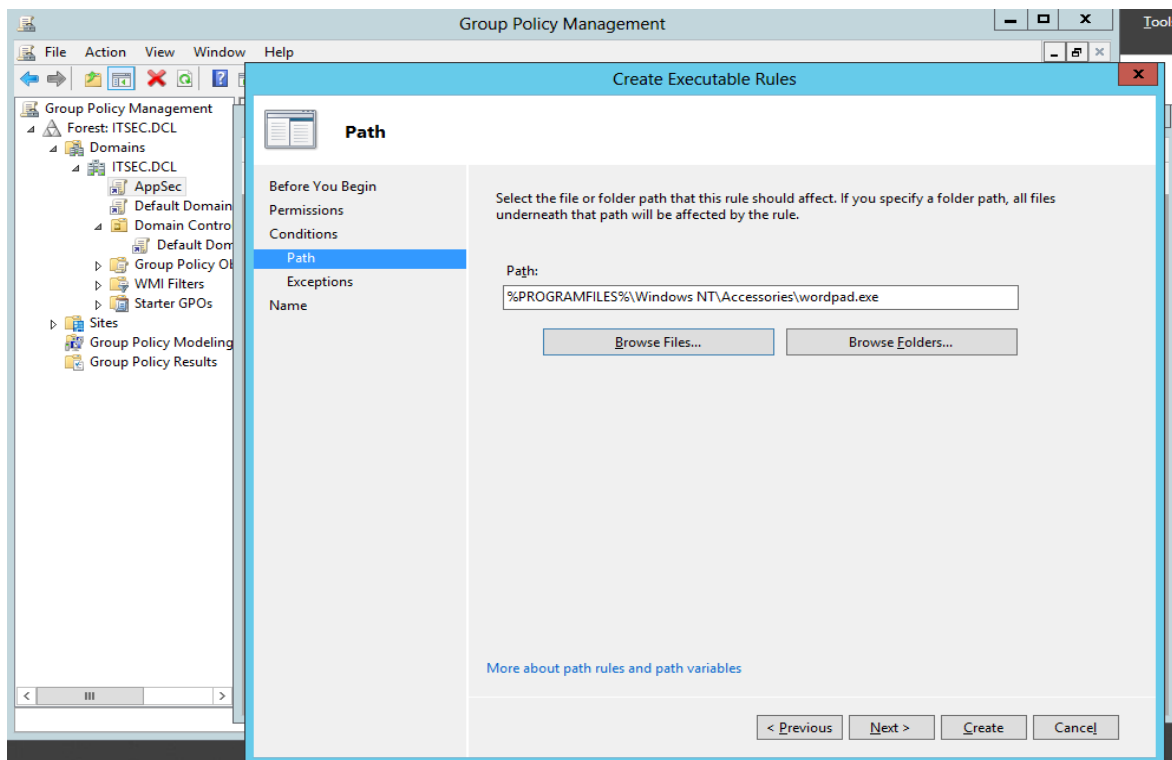
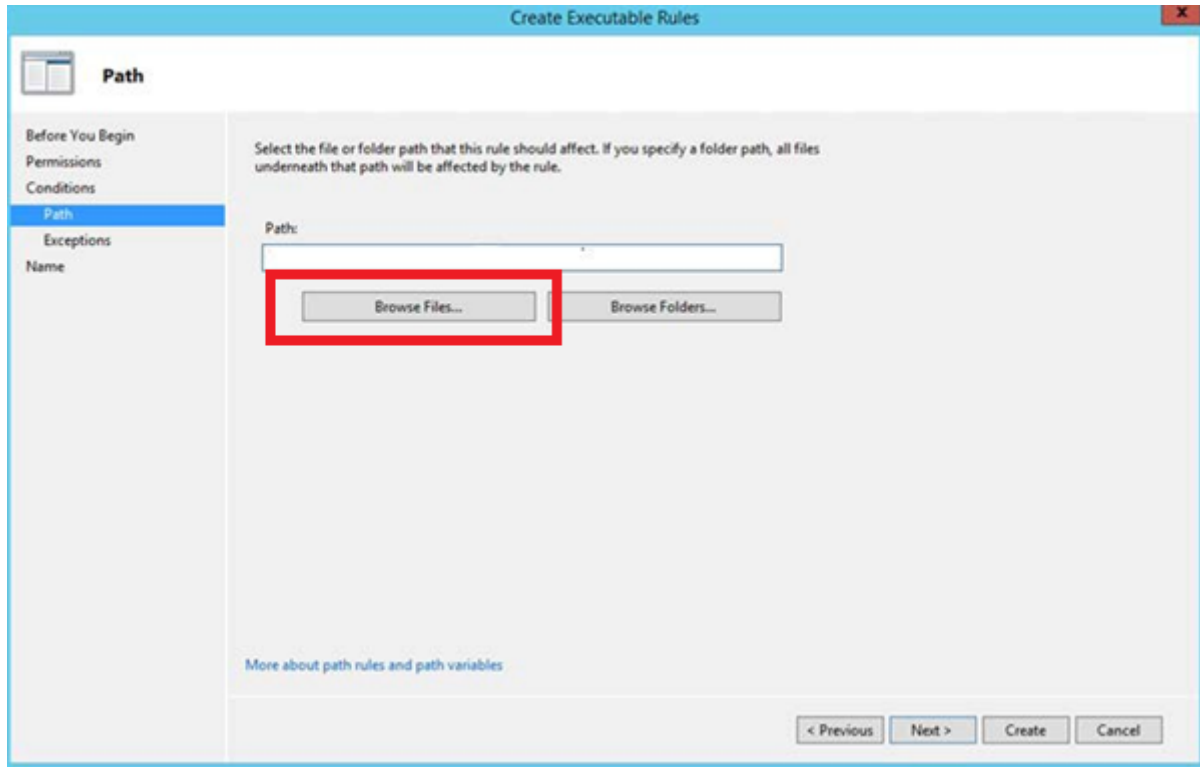
- **Publisher:** identify the applications signed by a specific publisher;
- **Path:** identify specific files and paths;
- **File Hash:** identify applications based on their digital

fingerprint. In our lab we chose *Path*:



The screenshot shows a window titled "Create Executable Rules" with a sidebar on the left and a main content area. The sidebar has a "Conditions" tab selected, with other tabs like "Before You Begin", "Permissions", "Path", "Exceptions", and "Name". The main content area has a heading "Conditions" and a sub-heading "Select the type of primary condition that you would like to create." Below this, there are three radio button options: "Publisher" (with a description: "Select this option if the application you want to create the rule for is signed by the software publisher."), "Path" (selected, with a description: "Create a rule for a specific file or folder path. If you select a folder, all files in the folder will be affected by the rule."), and "File hash" (with a description: "Select this option if you want to create a rule for an application that is not signed."). At the bottom right, there are four buttons: "< Previous", "Next >", "Create", and "Cancel". A link "More about rule conditions" is also present.

11. Specify the *Path* by using *Browse Files*. Look for the file **wordpad**. It is located in **C:\Program Files (x86)\Windows NT\Accessories**. The click **Next**.



12. You can add exceptions if you need them. But we will just click Next.

The screenshot shows the 'Create Executable Rules' dialog box with the 'Exceptions' tab selected. The left sidebar contains a tree view with the following items: 'Before You Begin', 'Permissions', 'Conditions', 'Path', 'Exceptions' (highlighted), and 'Name'. The main area has a title bar 'Create Executable Rules' and a close button. Below the title bar is a section titled 'Exceptions' with a list icon. The main content area contains the following text: 'To add an exception, select the type of exception and then click Add. Exceptions are optional and allow you to exclude files that would normally be included in the rule. To continue configuring this rule without adding an exception, click Next.' Below this text is a 'Primary condition:' label followed by the text 'C:\Users\%%USERNAME%%\Applications\app.exe'. Underneath is an 'Add exception:' label followed by a dropdown menu showing 'Publisher'. Below the dropdown is a table titled 'Exceptions:' with two columns: 'Exception' and 'Type'. The table is currently empty. To the right of the table are three buttons: 'Add...', 'Edit', and 'Remove'. At the bottom right of the dialog are four buttons: '< Previous', 'Next >', 'Create', and 'Cancel'.

Exceptions

Before You Begin
Permissions
Conditions
Path
Exceptions
Name

To add an exception, select the type of exception and then click Add. Exceptions are optional and allow you to exclude files that would normally be included in the rule. To continue configuring this rule without adding an exception, click Next.

Primary condition:
C:\Users\%%USERNAME%%\Applications\app.exe

Add exception:
Publisher

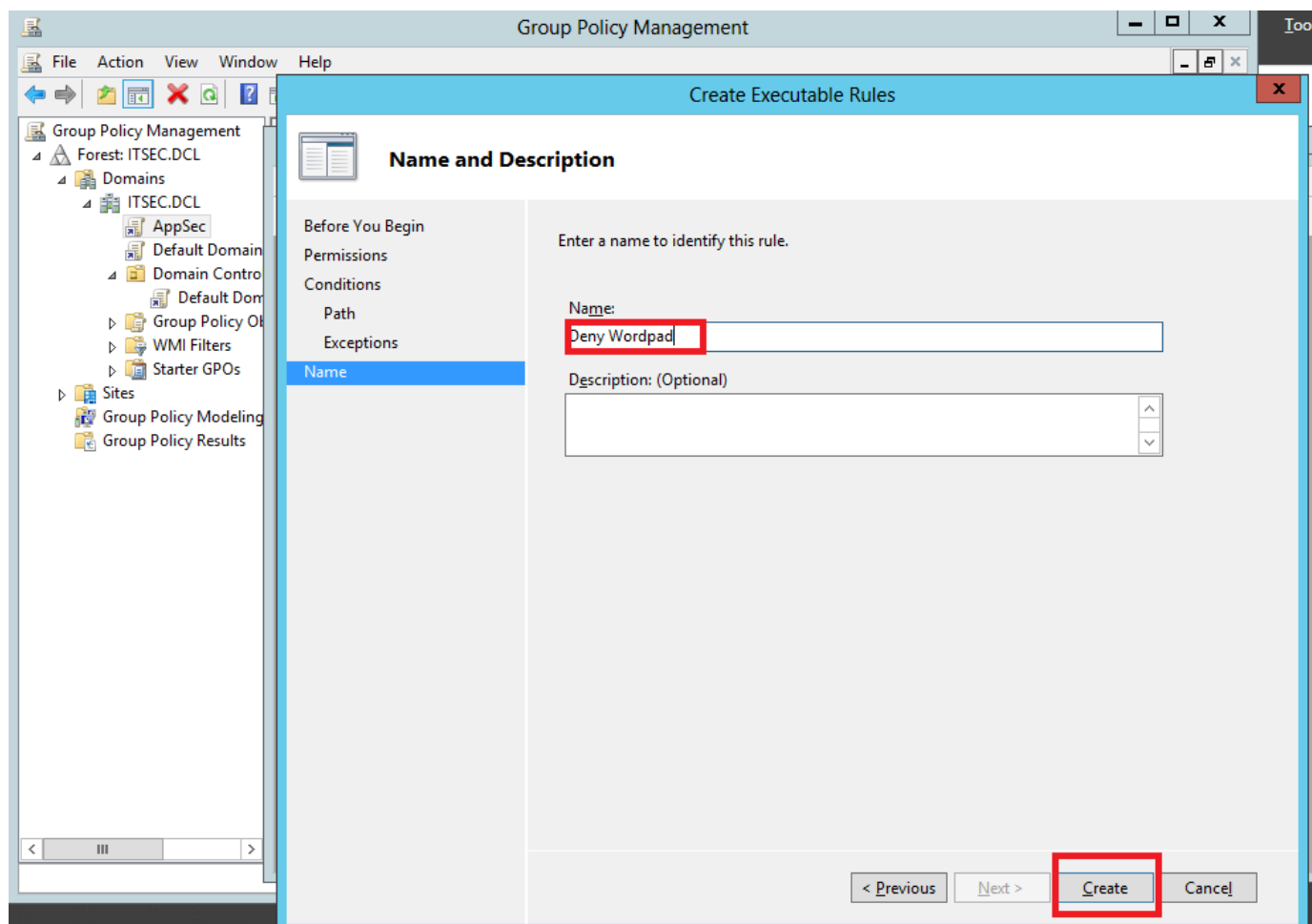
Exceptions:

Exception	Type
-----------	------

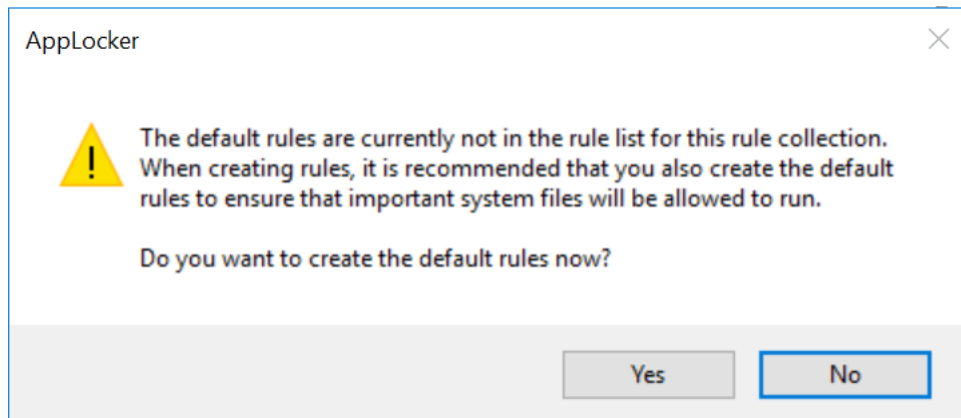
Add...
Edit
Remove

< Previous Next > Create Cancel

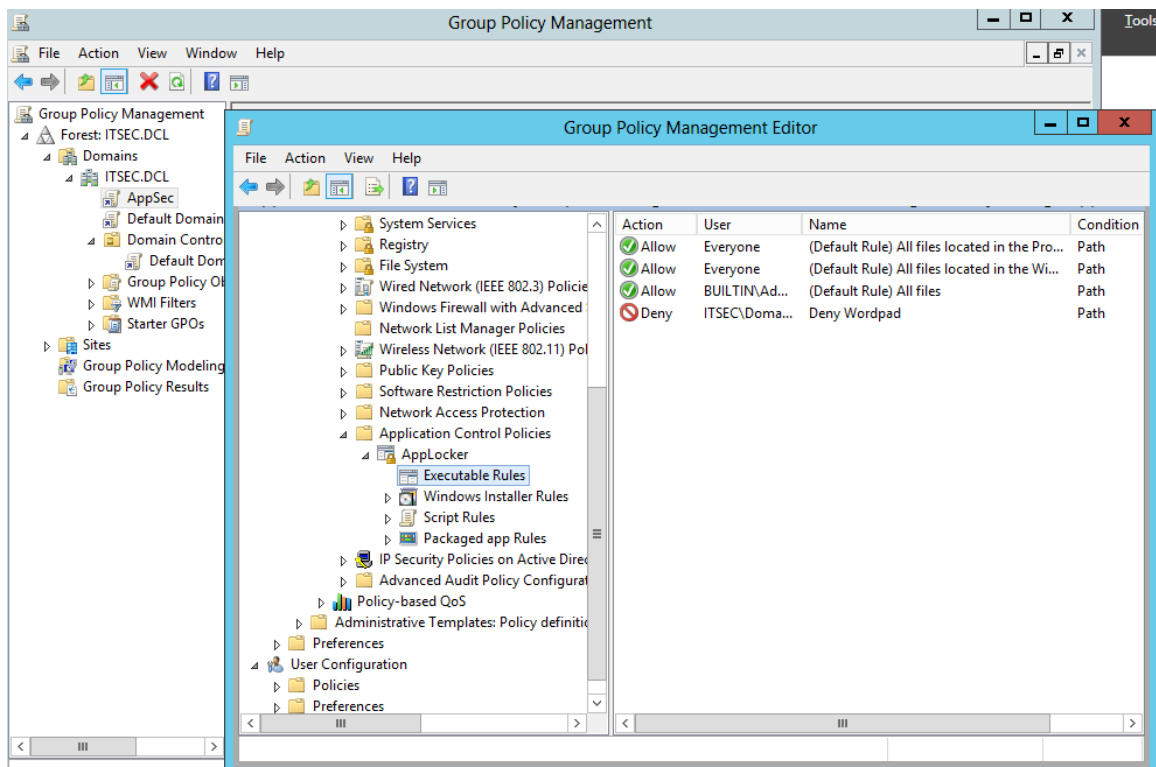
Name your "Deny Wordpad" and click **Create**:



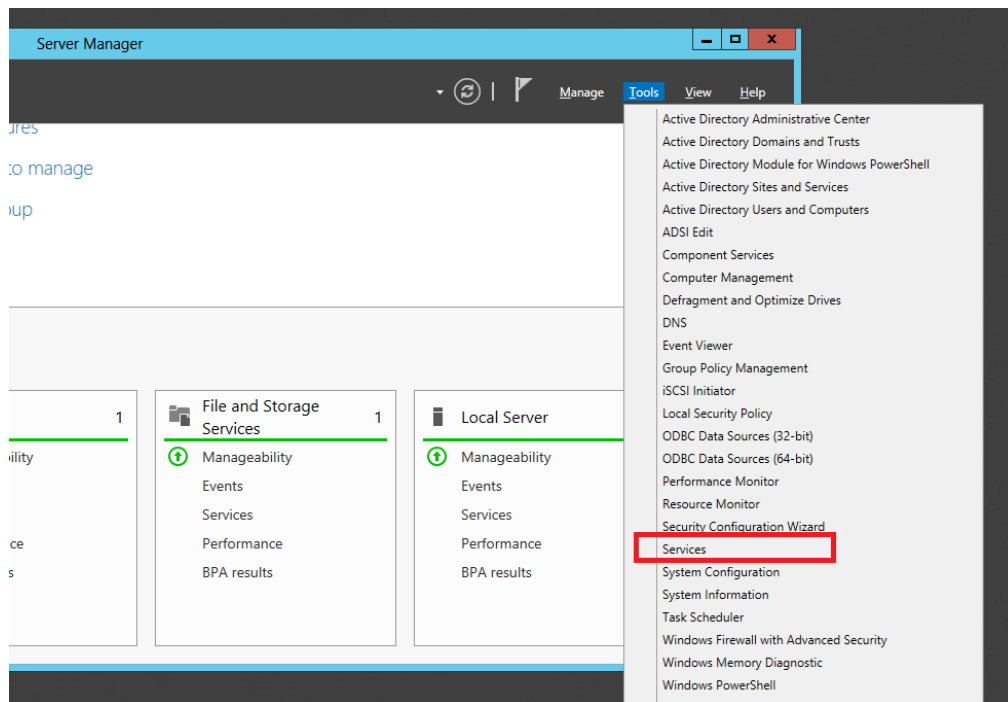
Click **Yes** if you see this screen.



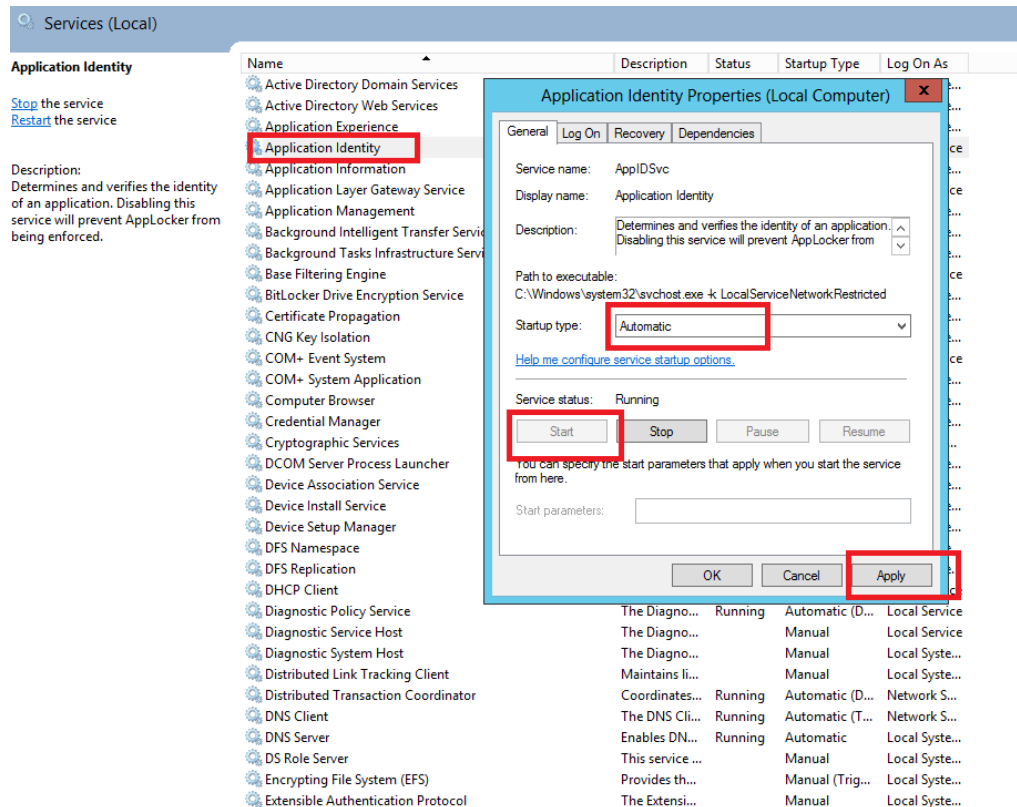
13. The rule will appear. Close the Group Policy Editor.



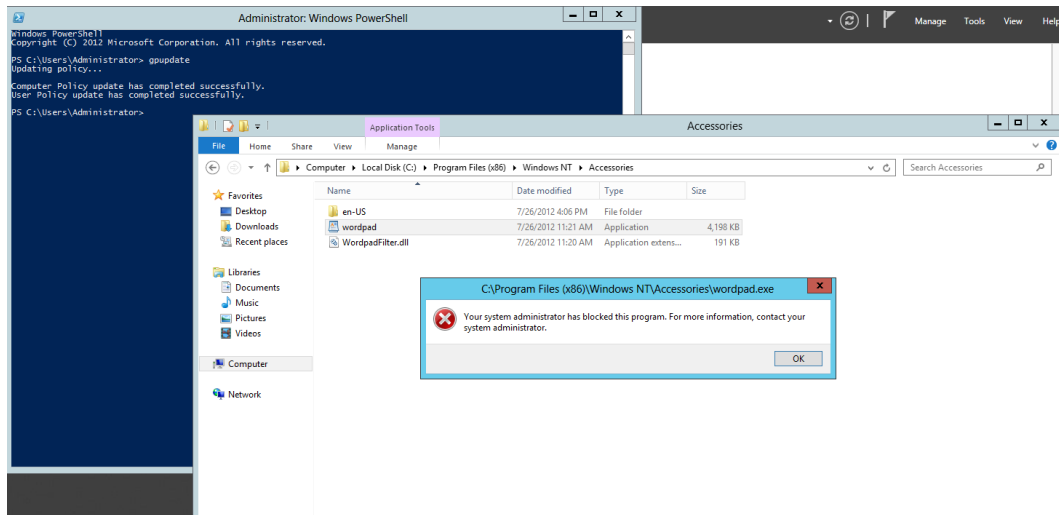
14. We can now test the rule. Go to **Server Manager** and Select **Services**.



15. Set the Startup type to **Automatic** and click on the **Start** button under Service status.. Click on **Apply** and then **OK**. Ignore the message “Access Denied” if it pops up.



16. Test the rule by running **gpupdate /force** using the Windows Shell and try running some Wordpad by opening Explorer and going to the path of Wordpad C:\Program Files (x86)\Windows NT\Accessories.
17. Restart the server if necessary.



[The End]