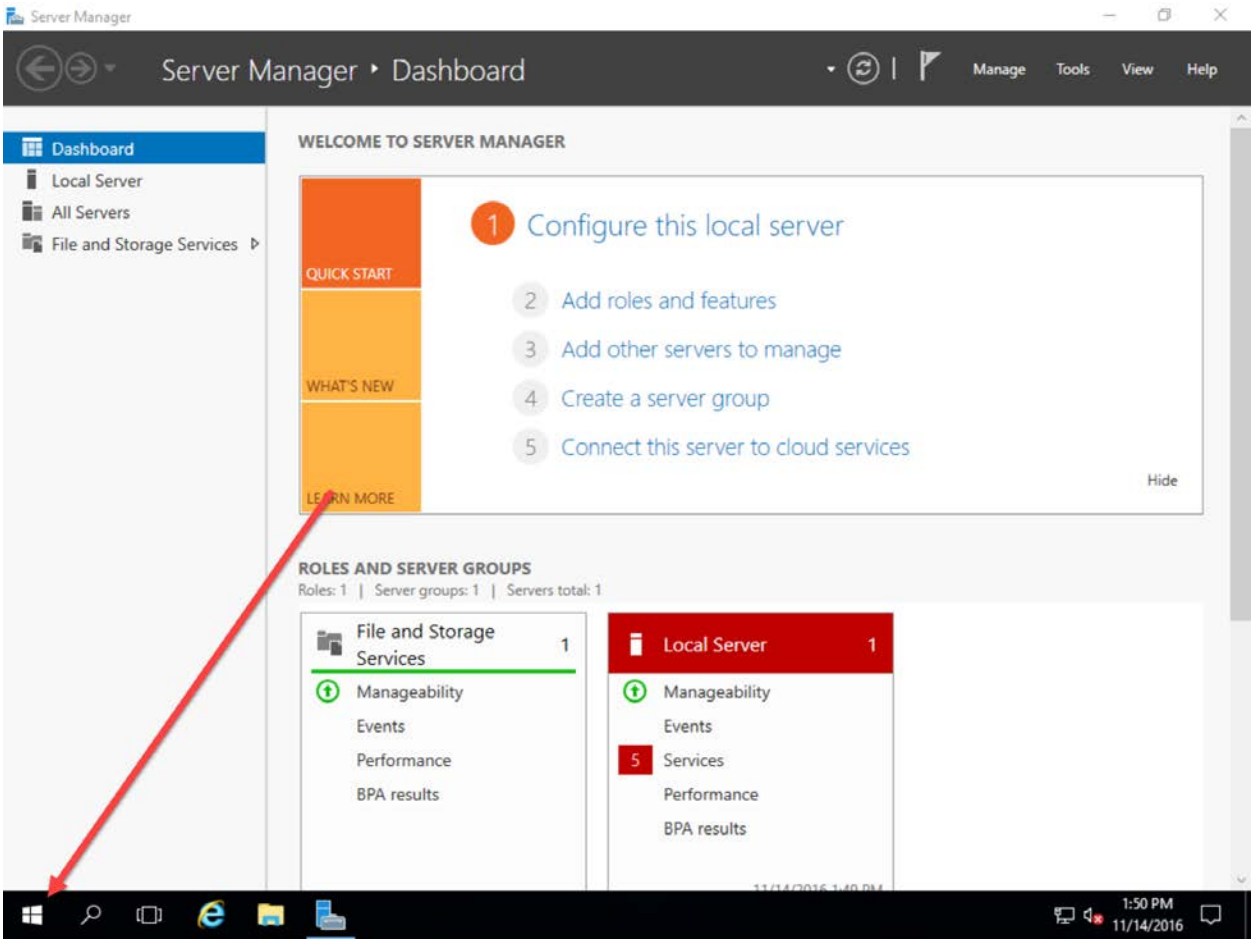
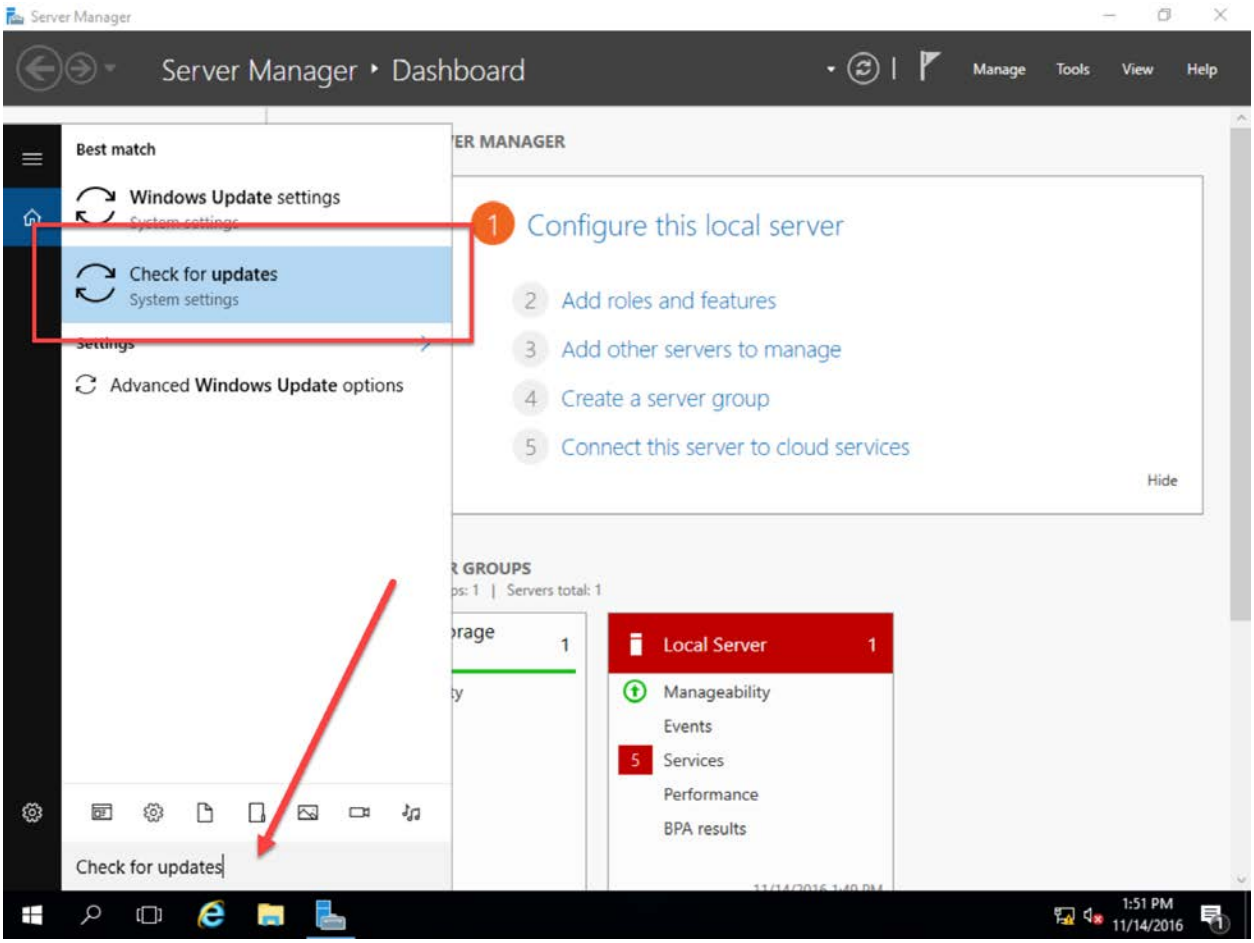
You should now be logged in and you should see the server manager dashboard. The next step is to run Windows Update to patch this server.



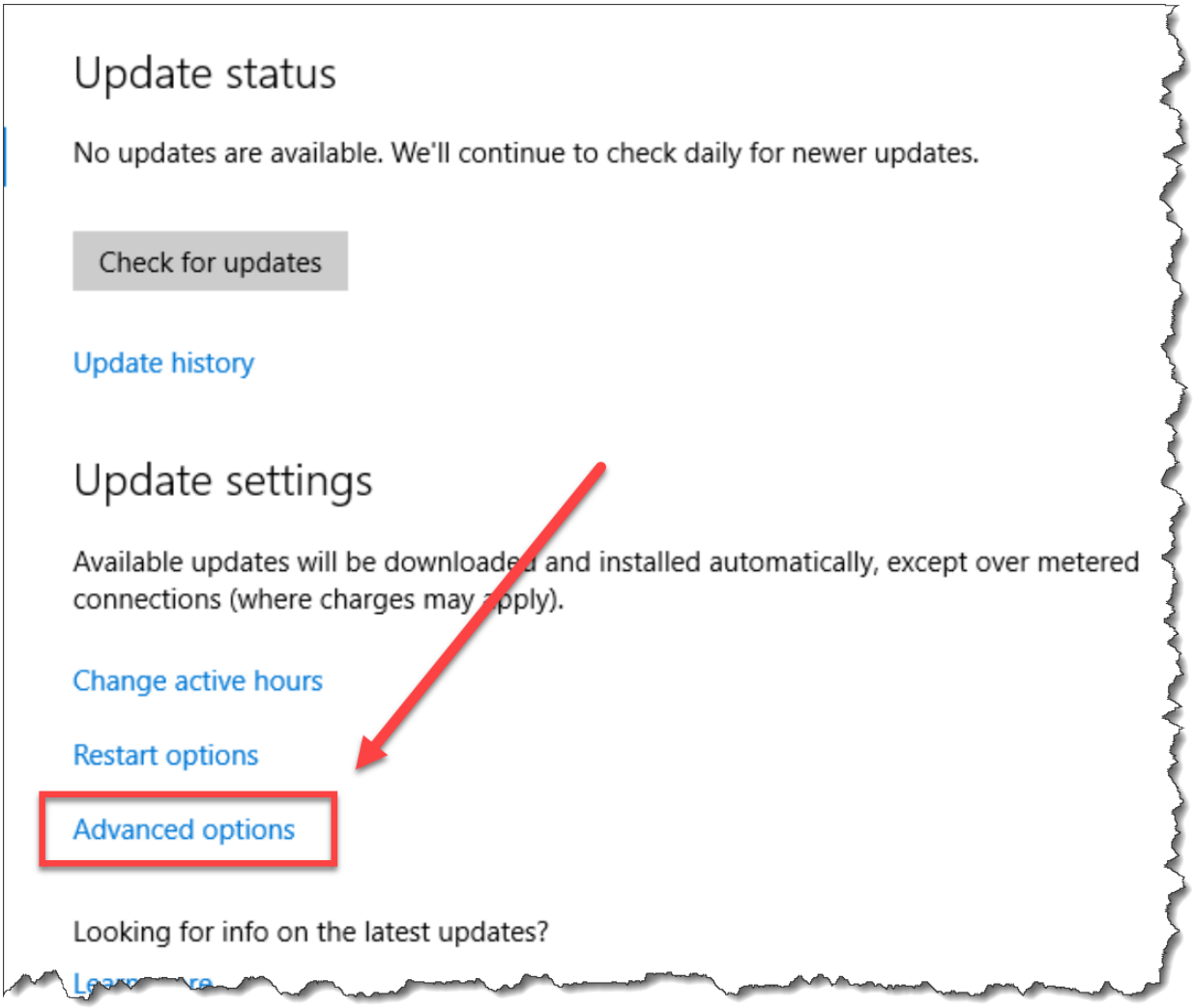
- Click the **Windows button** in the bottom left of the screen to navigate to the Start screen

You should now be at the Start menu.



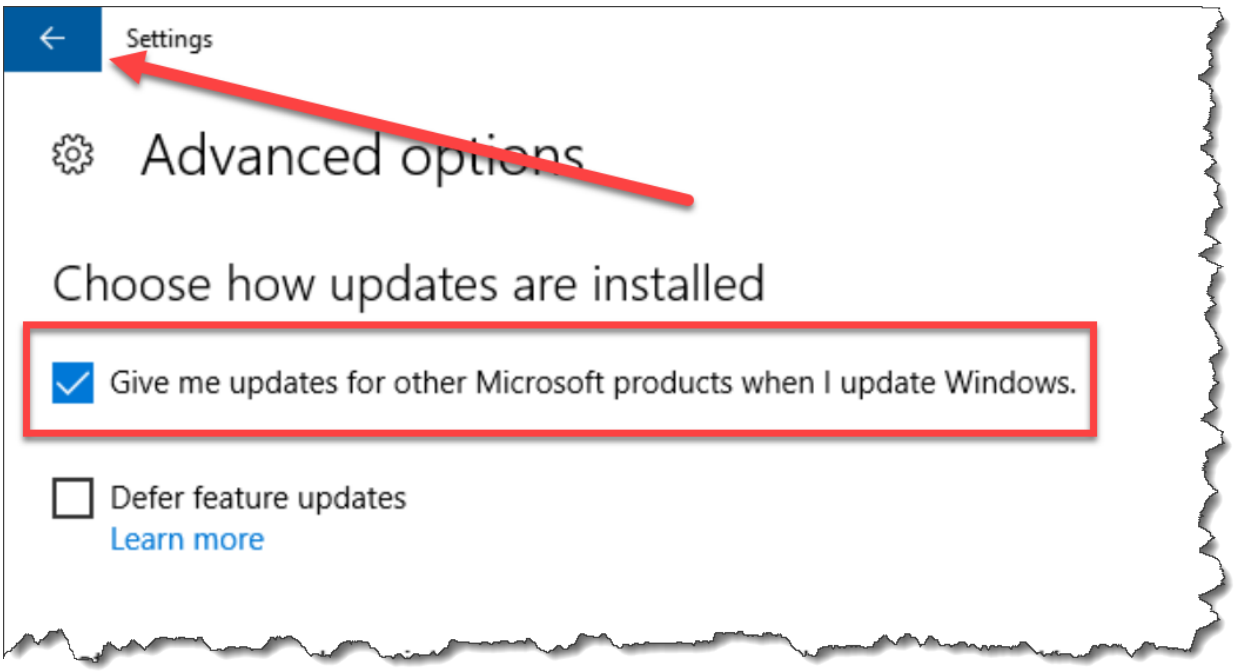
- In the search box, type **Check for updates**
- Click **Check for updates** in the search results

You should now see the Settings window. By default, Windows Update only gets patches for Windows itself but we want to enable patches for other products as well.



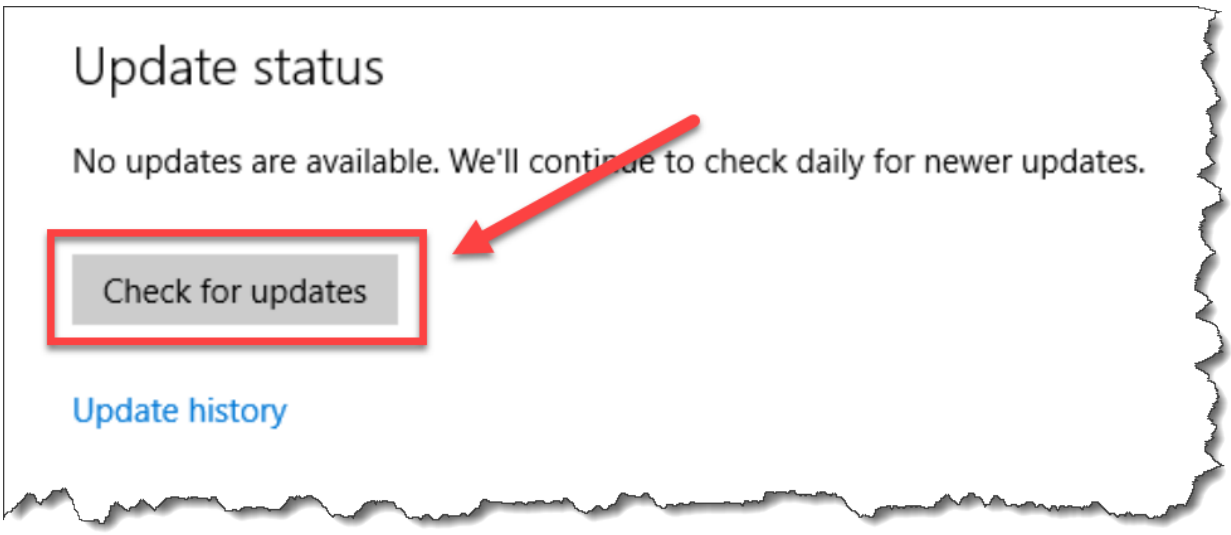
- Click the **Advanced options** link

You should be on a screen with the title **Advanced options**.



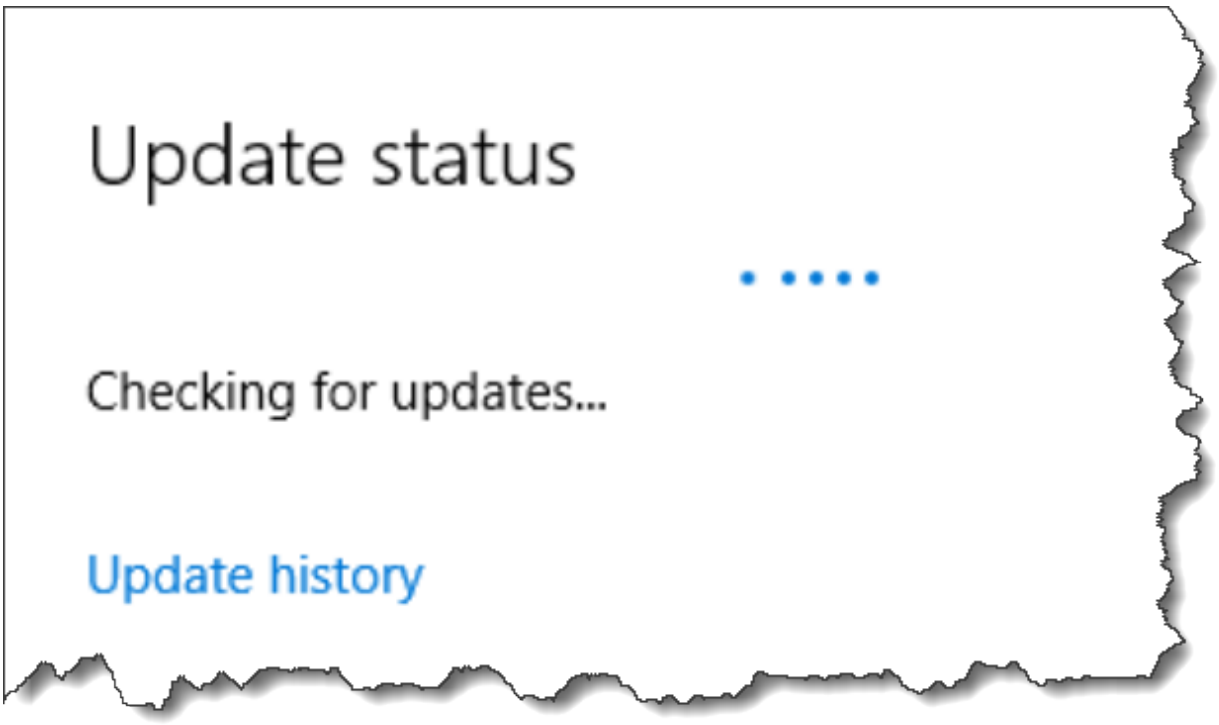
- Check **Give me updates for other Microsoft products when I update Windows**
- In the upper left corner of the screen, click the back arrow button

You should be back on the **Update status** screen.



- Click the **Check for updates** button

Windows Update should now be checking for updates.



Windows Update will probably find a ton of available updates. Let them all run and reboot your server as needed.

When the patches are all finished applying, make sure you're logged in as Administrator and continue on to the next page where I'll walk you through the optional setp of disabling a useless and annoying anti-feature in Windows called IE Enhanced Security.

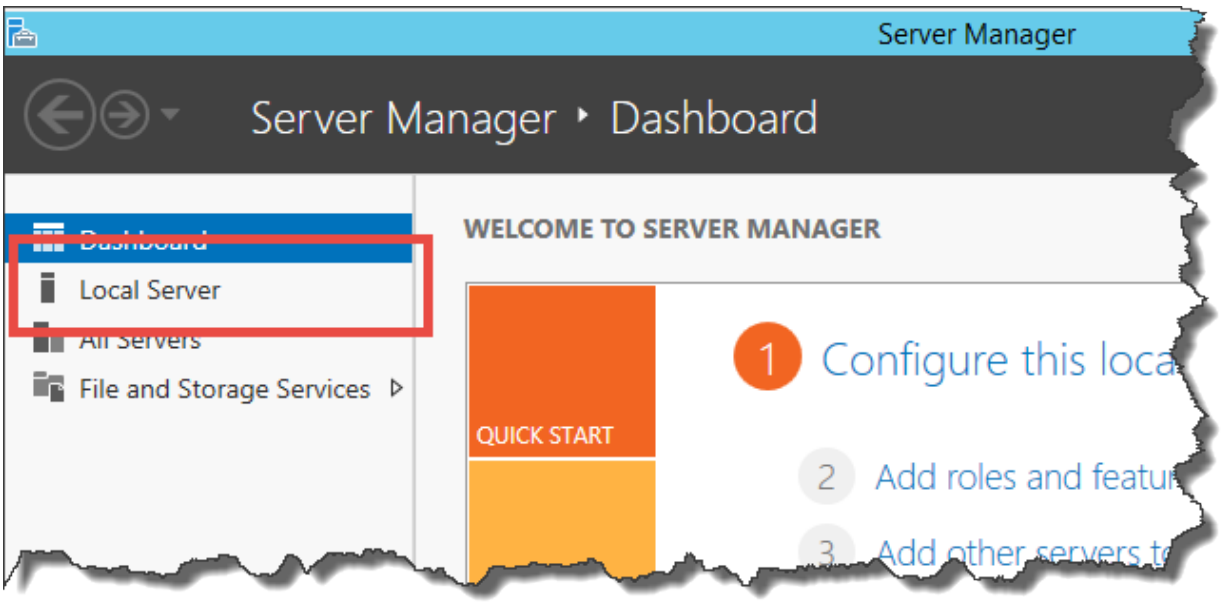
(Optional) Turn off IE Enhanced Security Configuration

Ok. So if you're paranoid about the internet and making sure that your server doesn't accidentally get infected by a virus that comes in because of someone's hapless browsing habits, you'll probably want to skip this section.

If you don't wear a tinfoil hat every day to keep the illuminati's evil space rays from controlling your mind, then you're probably like me and find IE Enhanced Security to be a royal pain in the behind. At some point, you're going to want to use the Internet Explorer web browser on this server and it'll be painful if IE Enhanced Security is turned on.

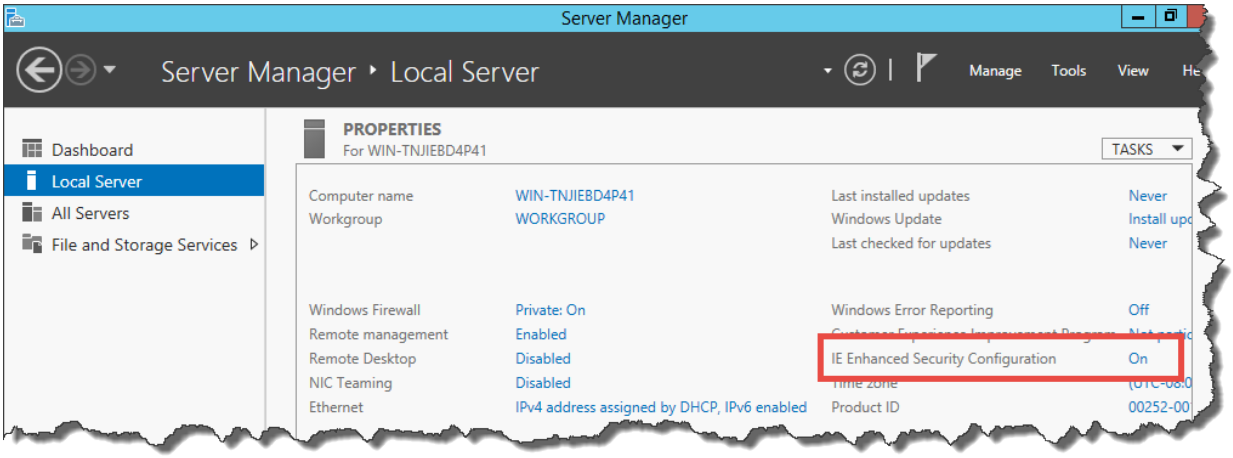
Let's turn it off.

You'll start this process by using **Server Manager**.



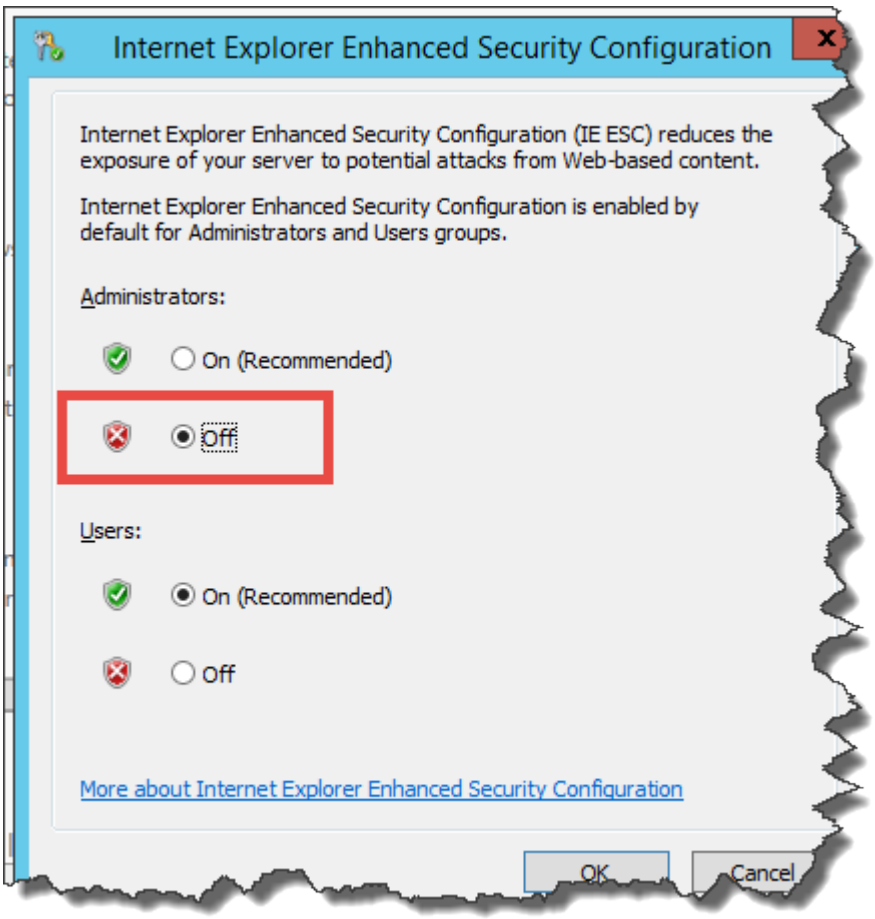
- In the left column of Server Manager, click **Local Server**

Towards the right side of the Server Manager window, you'll see an item that says **IE Enhanced Security Configuration**. It'll be set to **On**.



- Click the link that says **On**

You should now see the **Internet Explorer Enhanced Security Configuration** dialog.

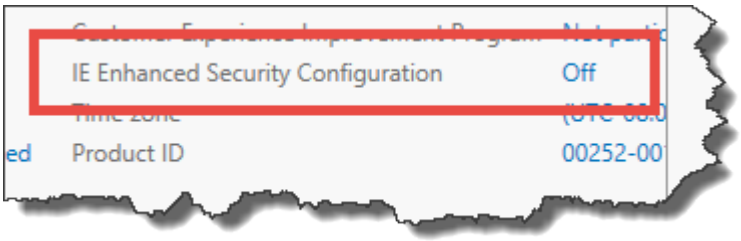


- Under Administrators, select the **Off** radio button
- (Optional) Under Users, select the **Off** radio button
- Click the **OK** button

You should now be back at the main page of the Server Manager.

- Click the **Refresh** button

IE Enhanced Security Configuration should now be set to Off.



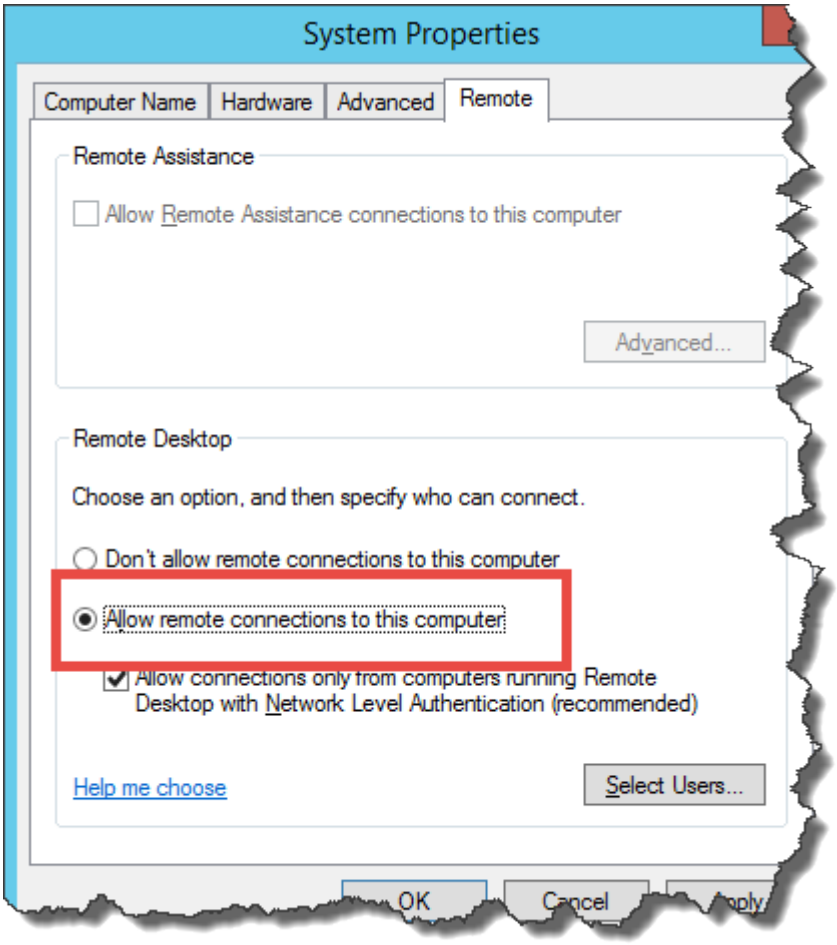
(Optional) Enable Remote Desktop

Are you lazy or maybe lazy-ish? Do you like convenience? Yah. Me, too. So that means that you'll probably want to enabled Remote Desktop. If you're a member of the Tinfoil Hat Patrol, you'll probably want to skip this section.

- Find **Remote Desktop** in Server Manager

- Click the **Disabled** link to the right of Remote Desktop

You should now see the **System Properties** dialog.



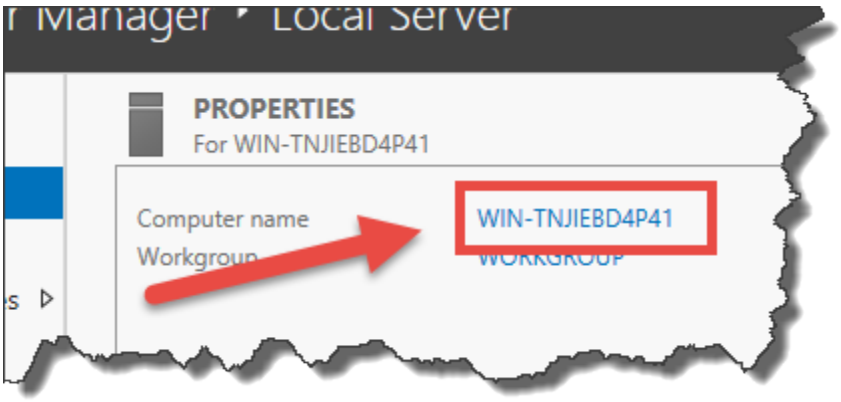
- In the Remote Desktop group, choose **Allow remote connections to this computer**
- Click the **OK** button

Remote Desktop is enabled.

Join this Server to the Active Directory Domain

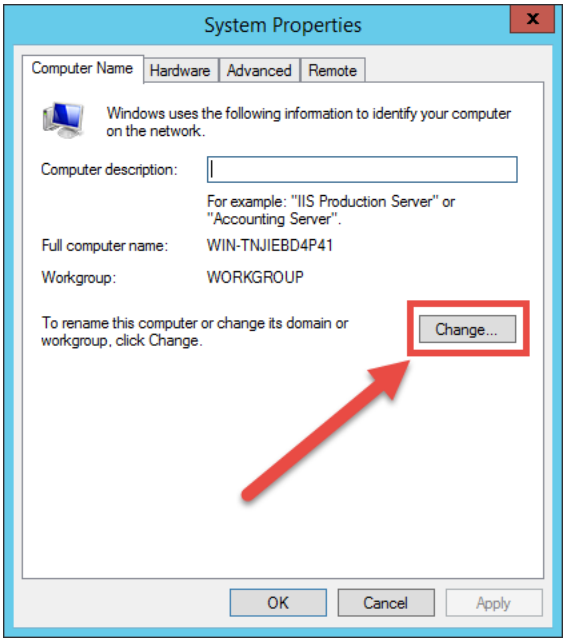
At the moment, you've got a stand-alone server with a wacky name that's not attached to anything. Workgroups? That's like the networking equivalent of having a stand-alone MP3 player that only syncs over USB. Who does that? What year is this?! 2005?! Not very useful. You'll now rename this computer and join it to your Active Directory domain so that it plays nicely with others.

In Server Manager, you'll see **Computer name** and **Workgroup**.



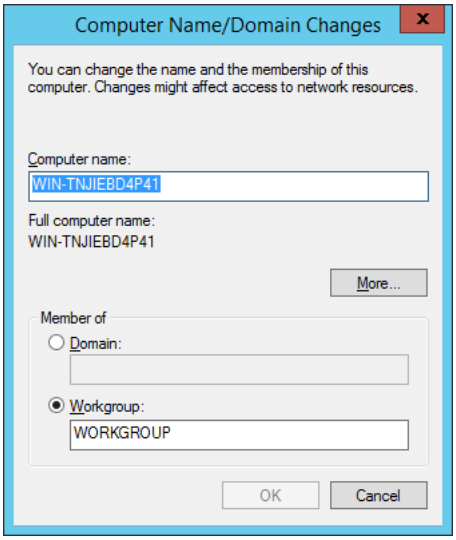
- Click on the computer name link

You should now see the **System Properties** dialog.

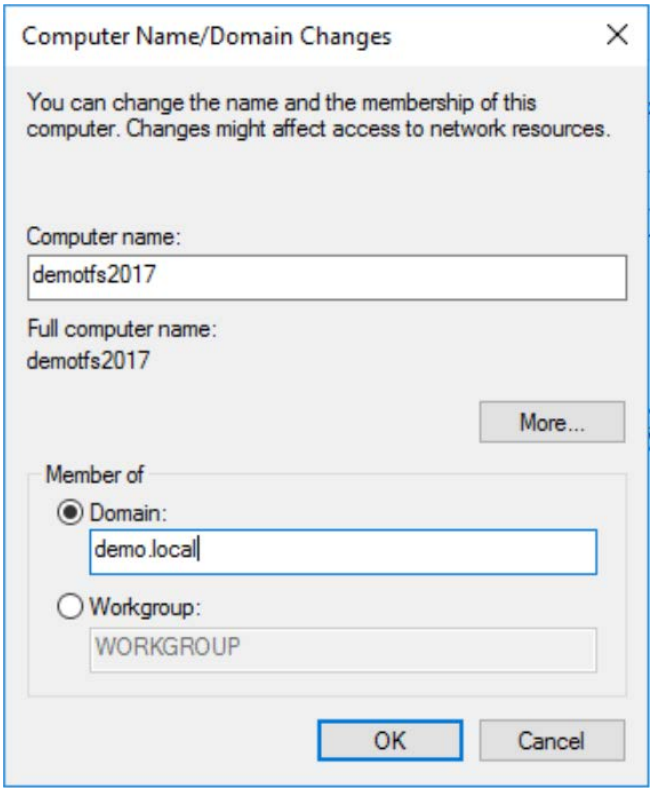


- Click the **Change...** button

You should now be on the **Computer Name/Domain Changes** dialog. The dialog should be showing you the current name of the computer and the workgroup membership.

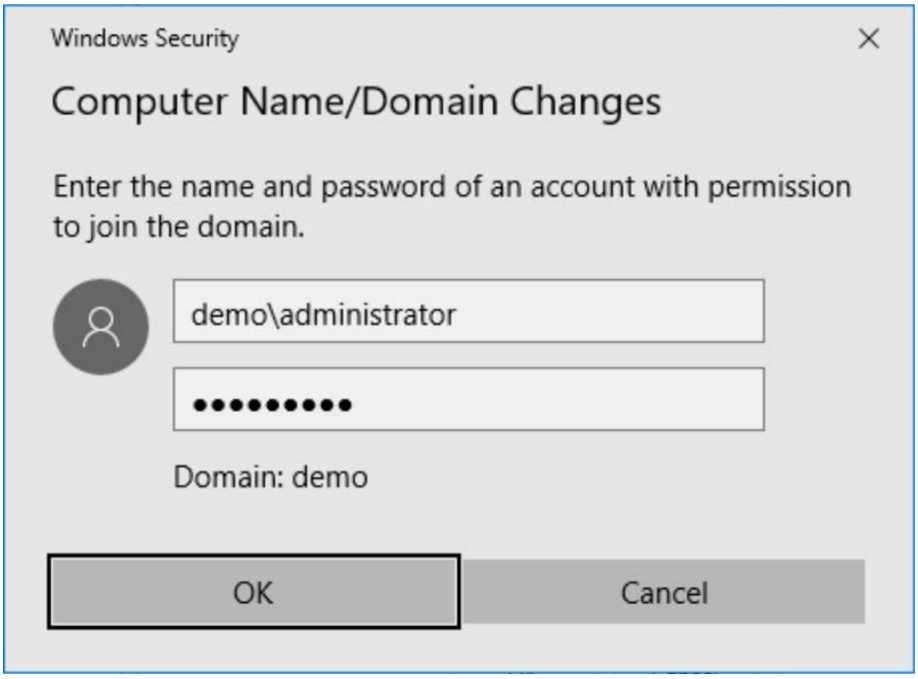


You should now change the values to be what you want the server to be named and the Active Directory domain that it should be attached to.



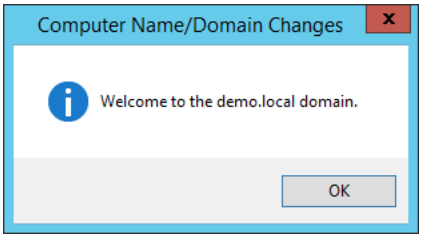
- In the Computer name textbox, enter the desired name for this server
- Under **Member of** choose the **Domain** radio button
- In the **Domain** textbox, enter the name of the Active Directory domain
- Click the **OK** button

You'll be prompted for the username and password for a domain administrator for the target domain.



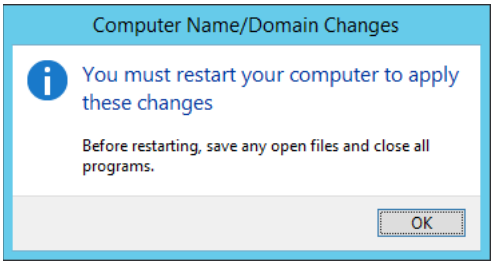
- Enter the username and password
- Click **OK**

You should see a dialog welcoming you to the new domain.

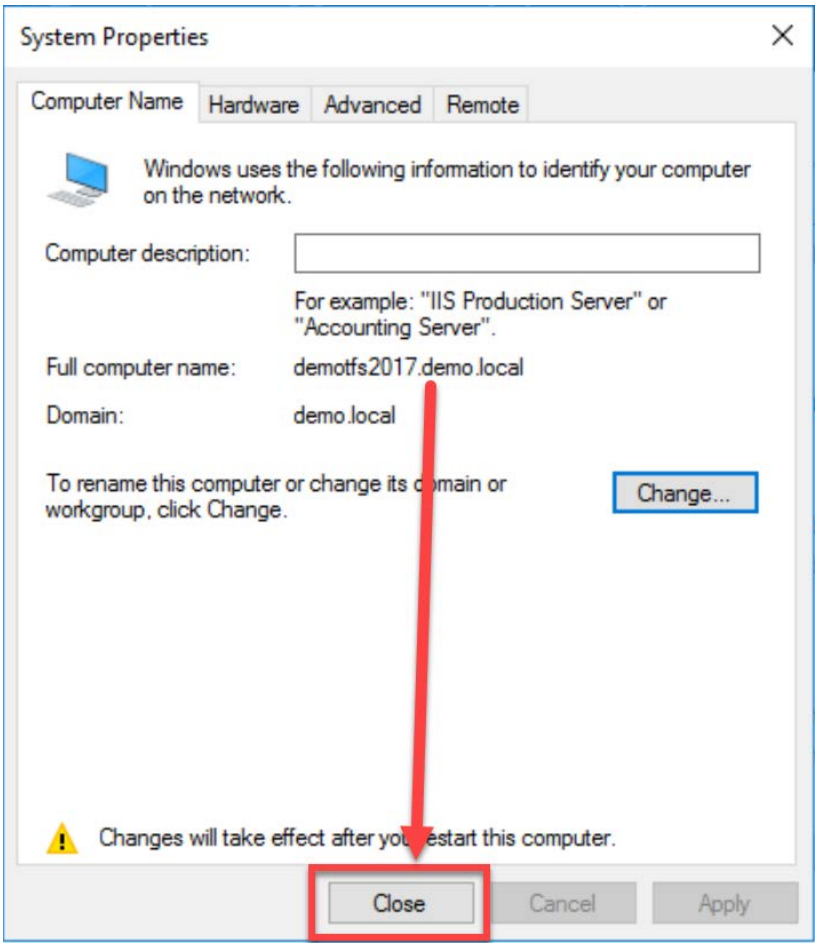


- Click **OK**

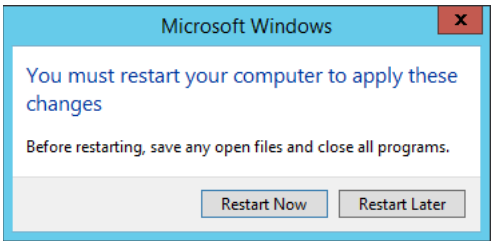
You'll be notified that you'll need to reboot this server.



- Click the **OK** button



- Click the **Close** button



- Click the **Restart Now** button

The server will restart and return you to the lock screen. The server has been installed and joined to the domain.

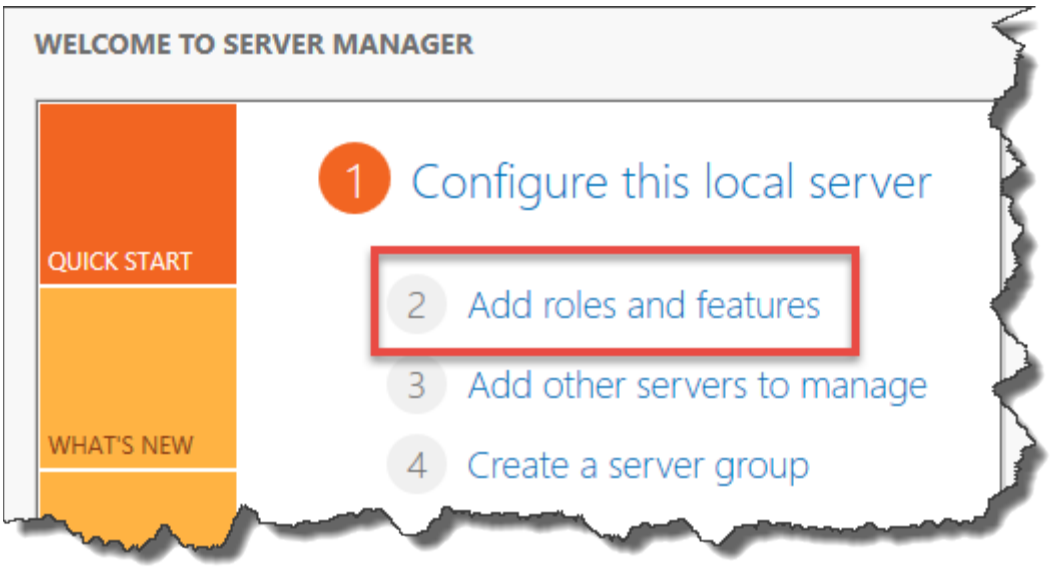
NOTE: It is *optional* but you might find it helpful to configure this server to have a static IP address and a static A record entry in your DNS server.

Chapter 2: Install Pre-requisites for SQL Server 2016 and Team Foundation Server 2017

Before you can install SQL Server or Team Foundation Server, you'll need to enable their pre-requisite roles and features in Windows Server.

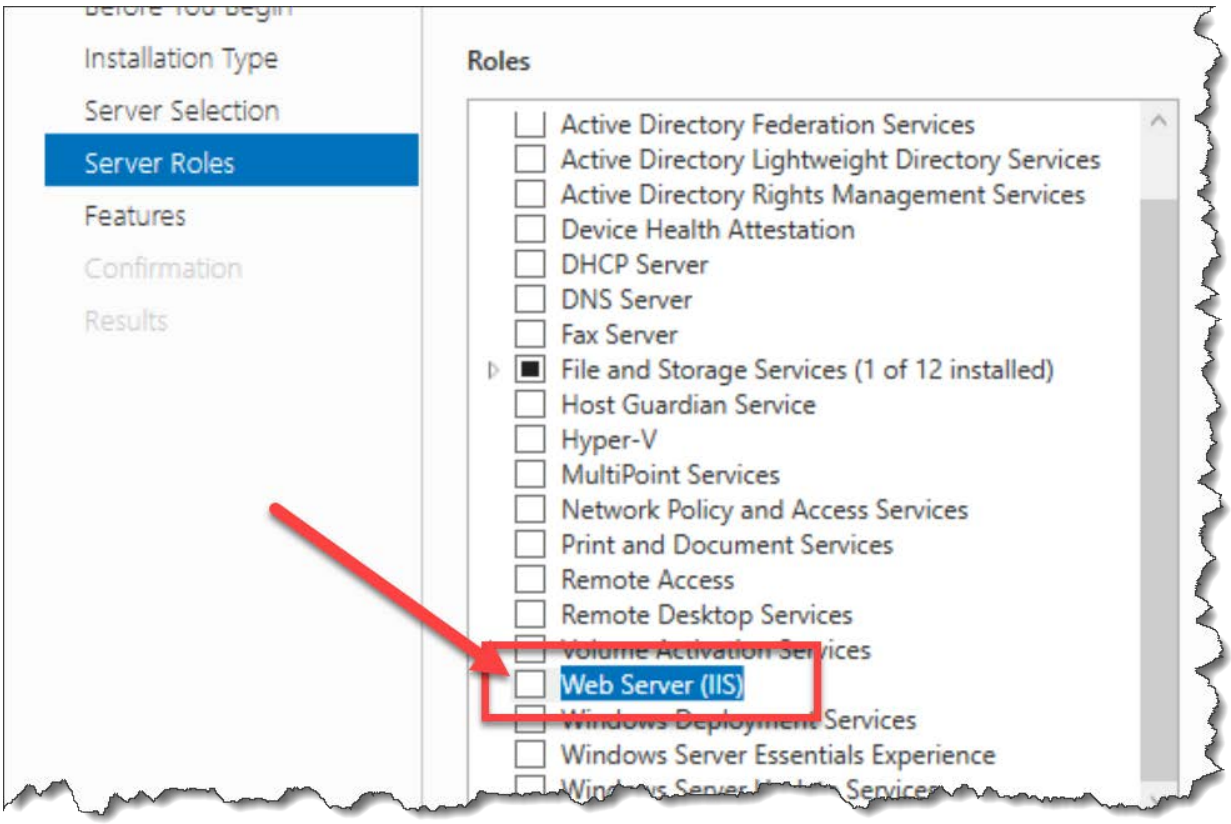
- Log on to the server using an account that is a member of the **Administrators** group
- Run **Server Manager**

First we need to verify that the .NET Framework 3.5 features are installed on this server.



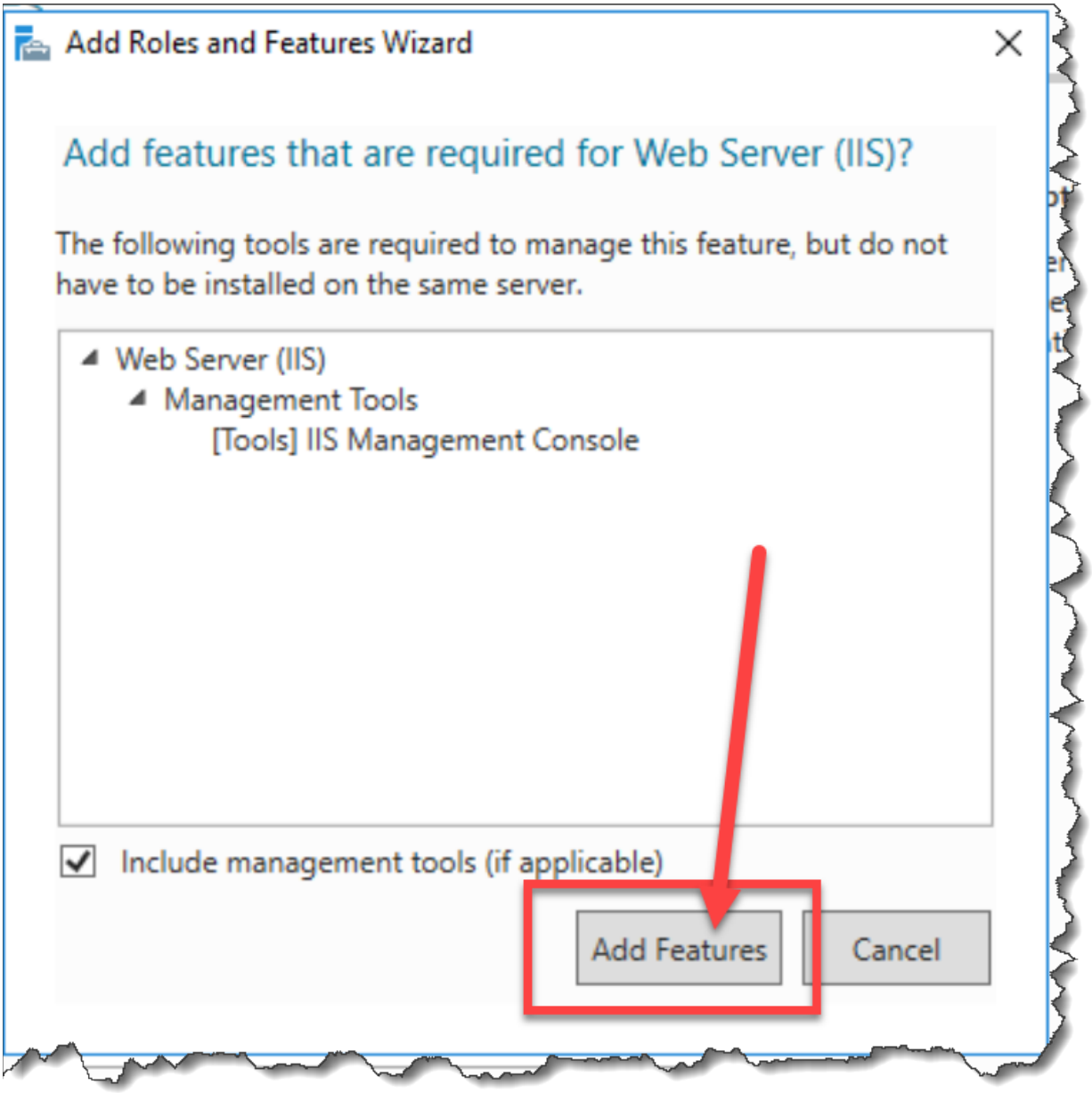
- In Server Manager, click **Add roles and features**
- On the **Before you begin** page of the wizard, click **Next**
- On the **Select installation type** page, choose **Role-based or feature-based installation** and click **Next**
- On the **Select destination server** page
  - Choose **Select a server from the server pool**
  - Select the name of the current server
  - Click **Next**

You should now be on the **Select server roles** page of the wizard.



- Check **Web Server (IIS)**

You'll see an **Add Roles and Features Wizard** dialog prompting you to add some additional features.

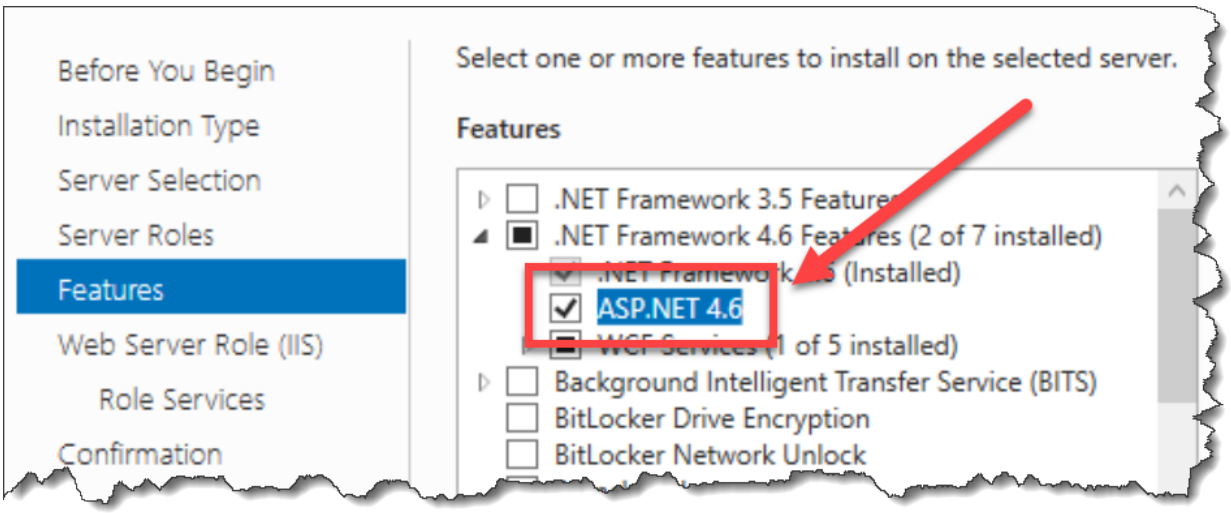


- Click **Add Features**

Web Server (IIS) should now be checked.

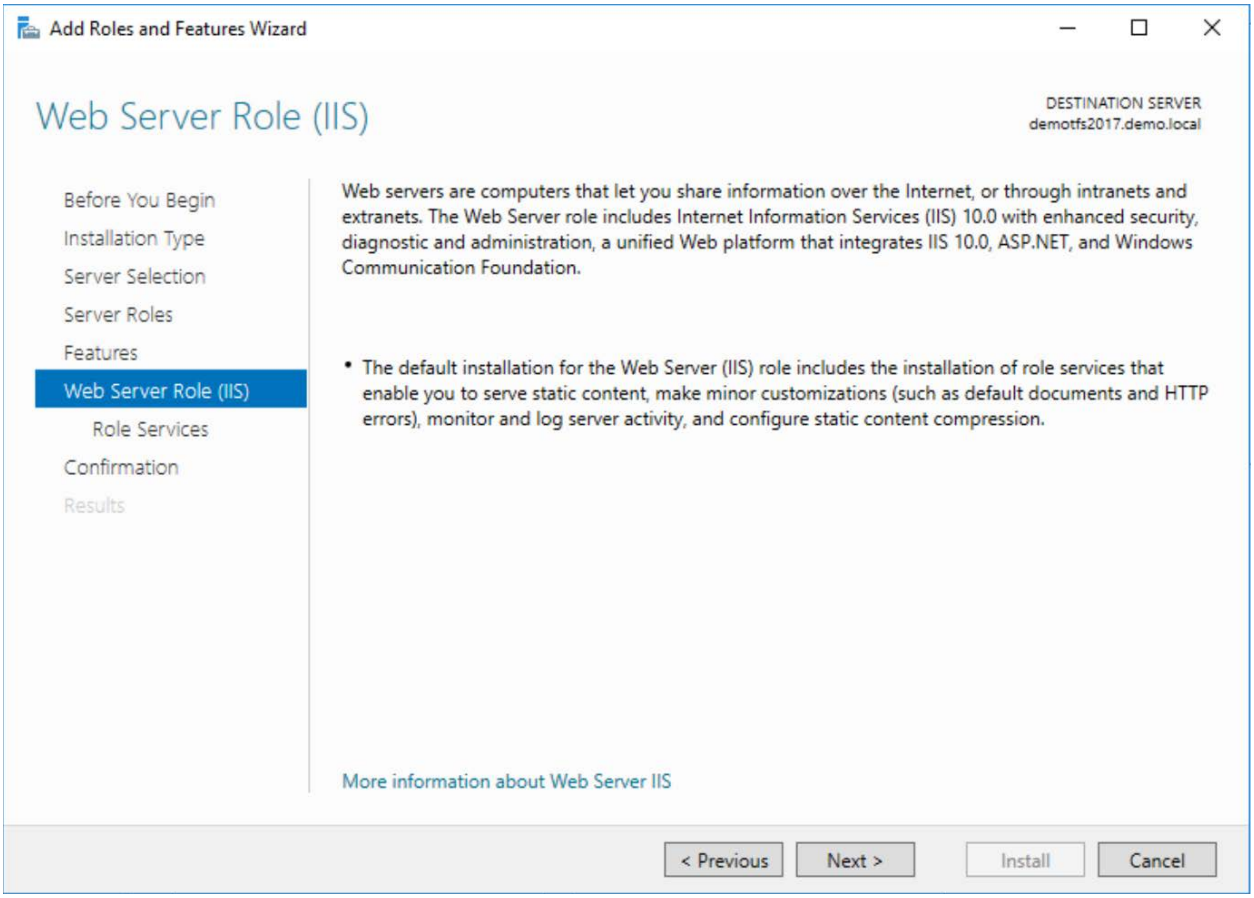
- Click **Next**

You should now be on the **Select features** page.



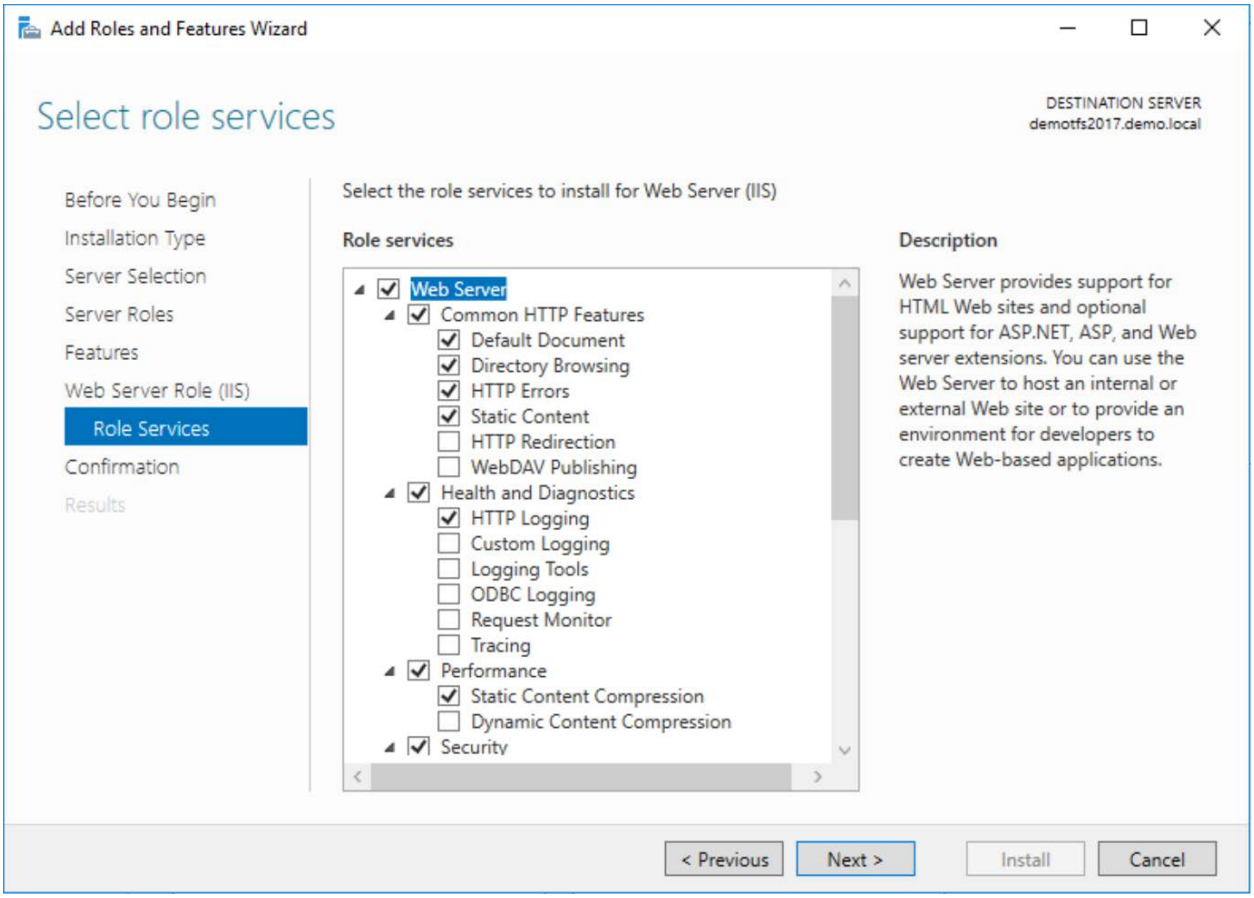
- Verify that **.NET Framework 4.6 Features** is checked
- Expand the .NET Framework 4.6 Features node
- Check **ASP.NET 4.6**
- Click **Next**

You'll probably see a message about the Web Server (IIS) role.



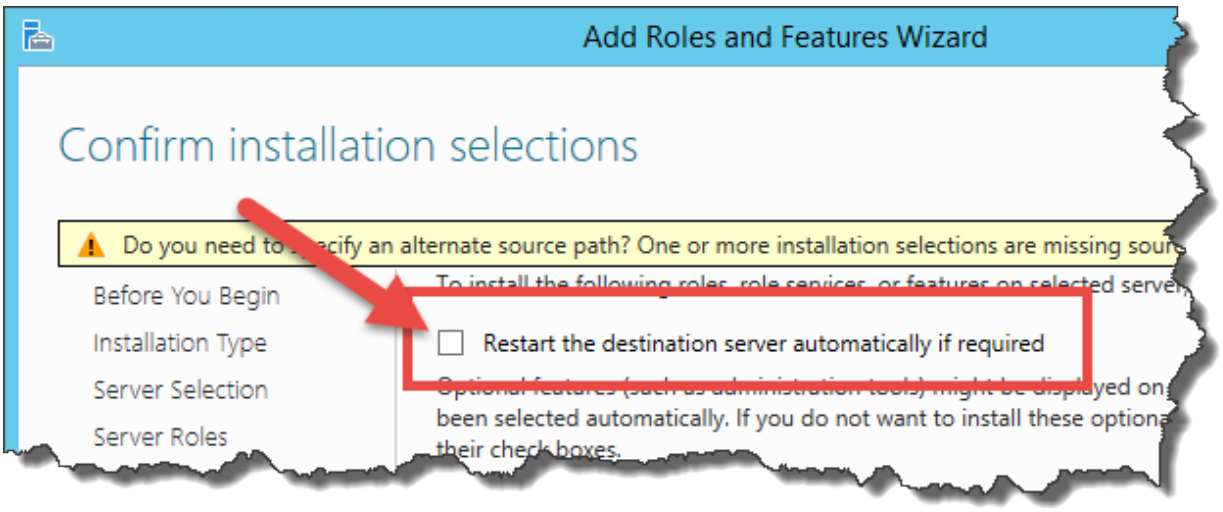
- Click **Next**

You should now see the **Select role services** page of the wizard.



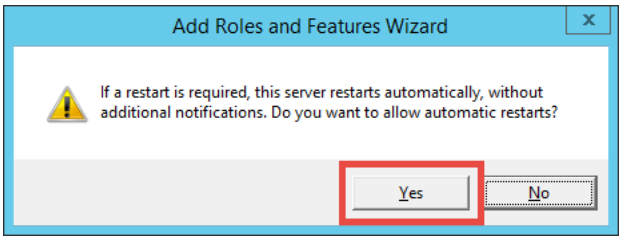
- Click the **Next** button

You should now be on the **Confirm installation selections** page.



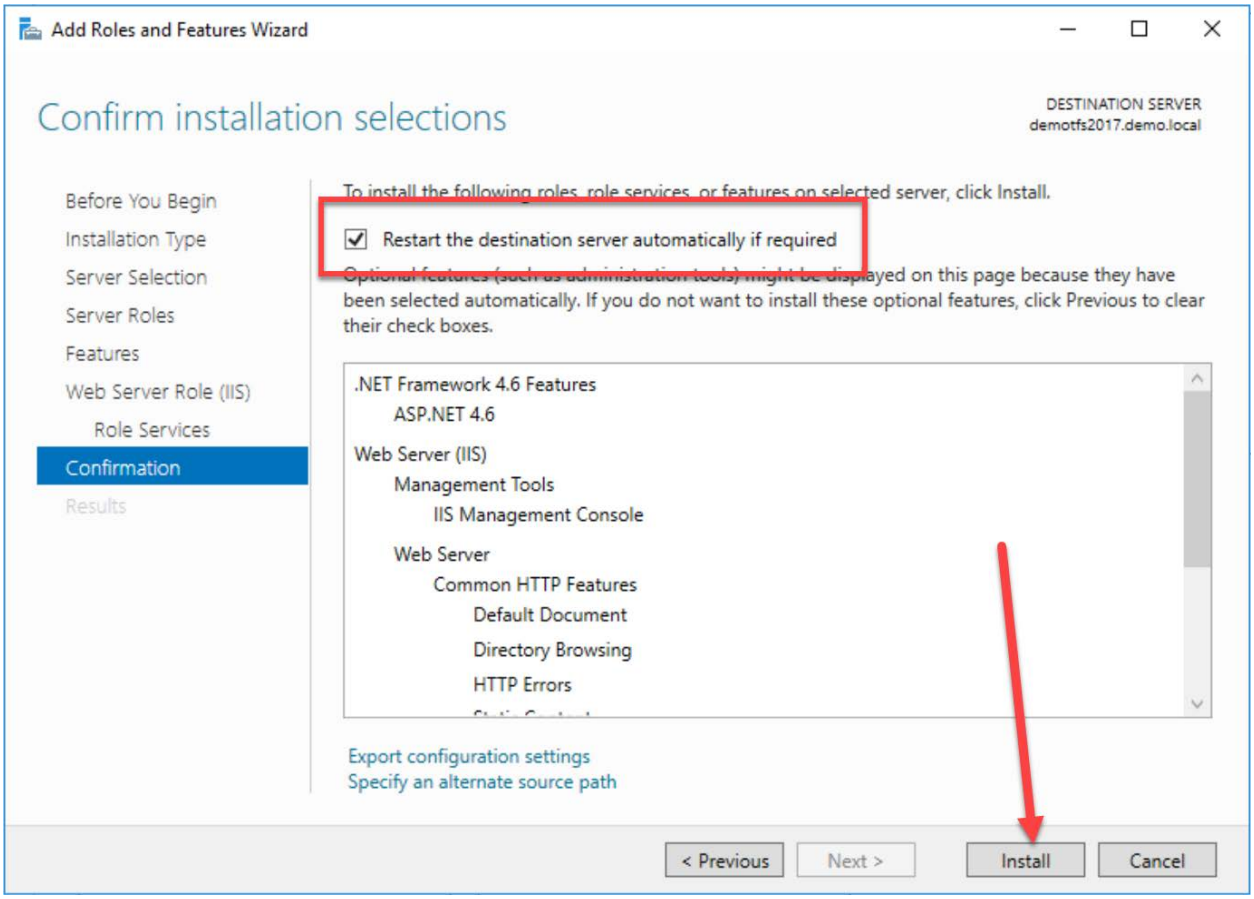
- Click the checkbox for **Restart the destination server automatically if required**

You'll be prompted to approve automatic restarts.



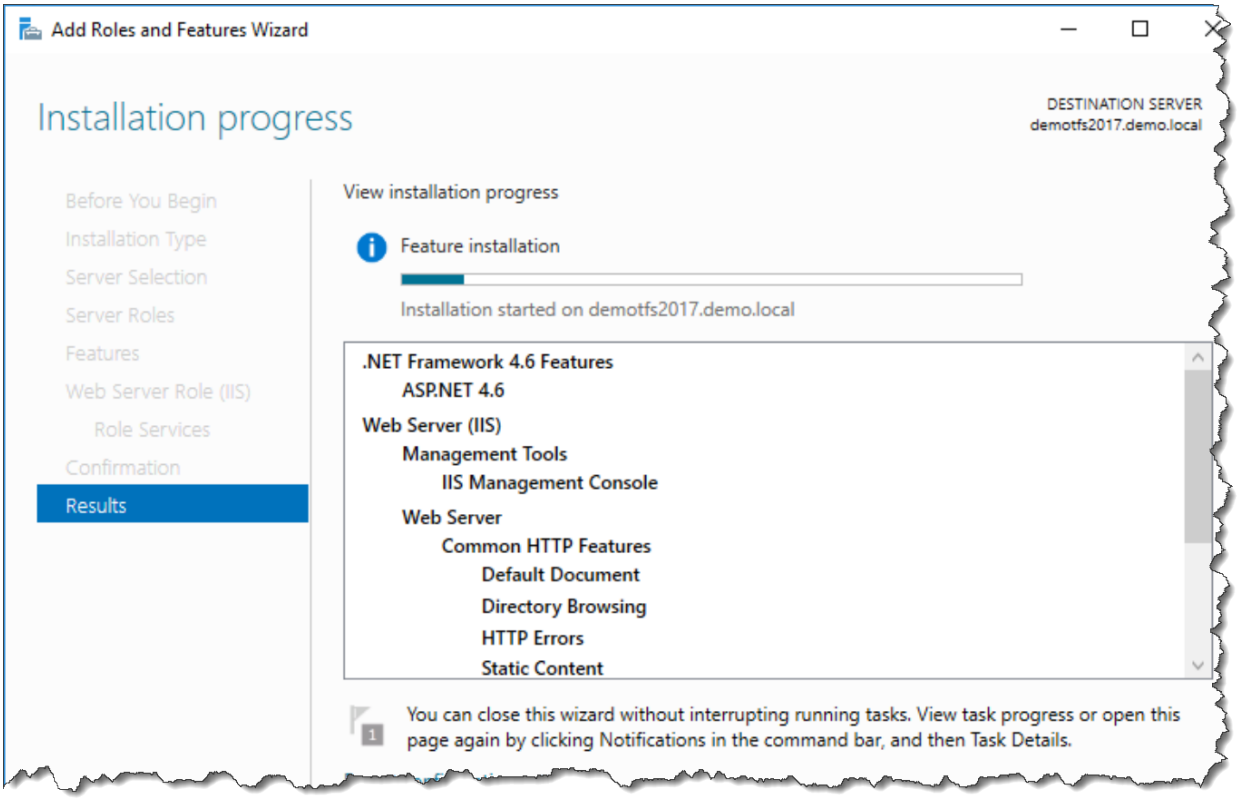
- Click **Yes**

You should now be on the **Confirm installation selections** page and **Restart the destination server automatically if required** should be checked.

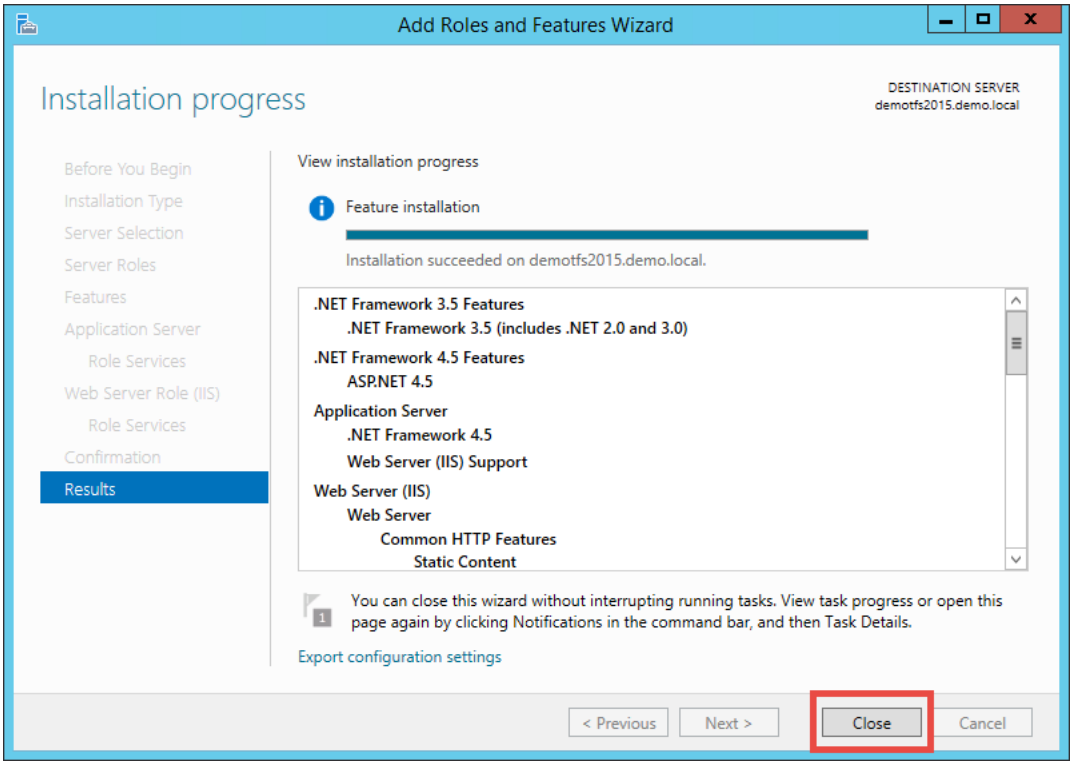


- Check **Restart the destination server automatically if required**
- Click **Install**

The features should now be installing.



Eventually, the feature installation should finish.

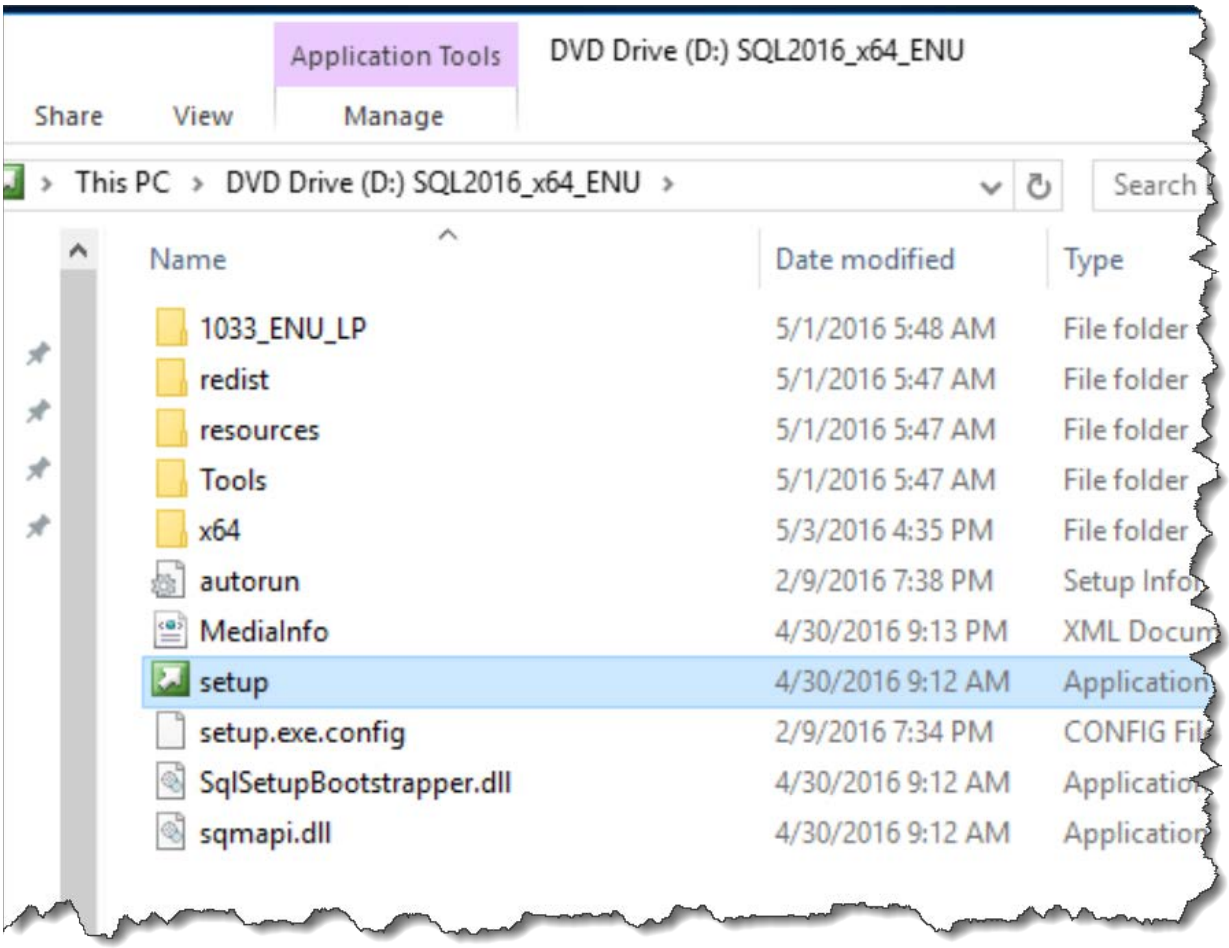


- Verify that the installation succeeded
- Click **Close**
- (Optional) **Reboot** the server

Chapter 3: Install SQL Server 2016

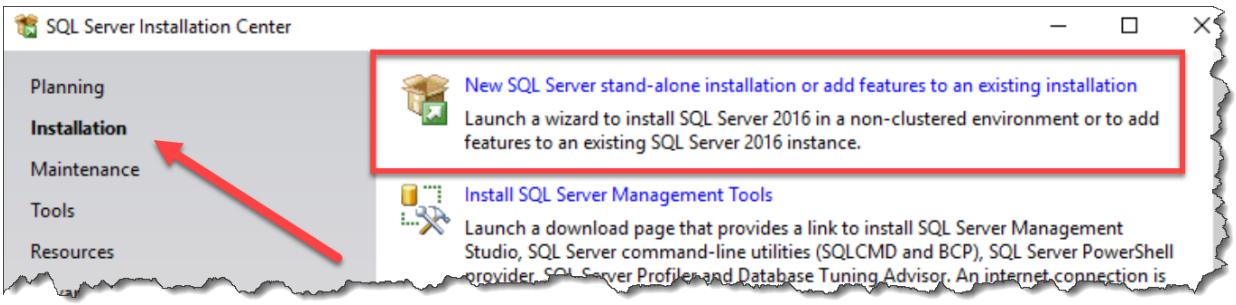
This chapter will walk you through the process of installing SQL Server 2016 for use with TFS.

- Log on to the server using an account that is a member of the **Administrators** group
- Insert the SQL Server 2016 DVD or mount the appropriate ISO image
- Open the DVD drive using **Windows Explorer** (**explorer.exe)**



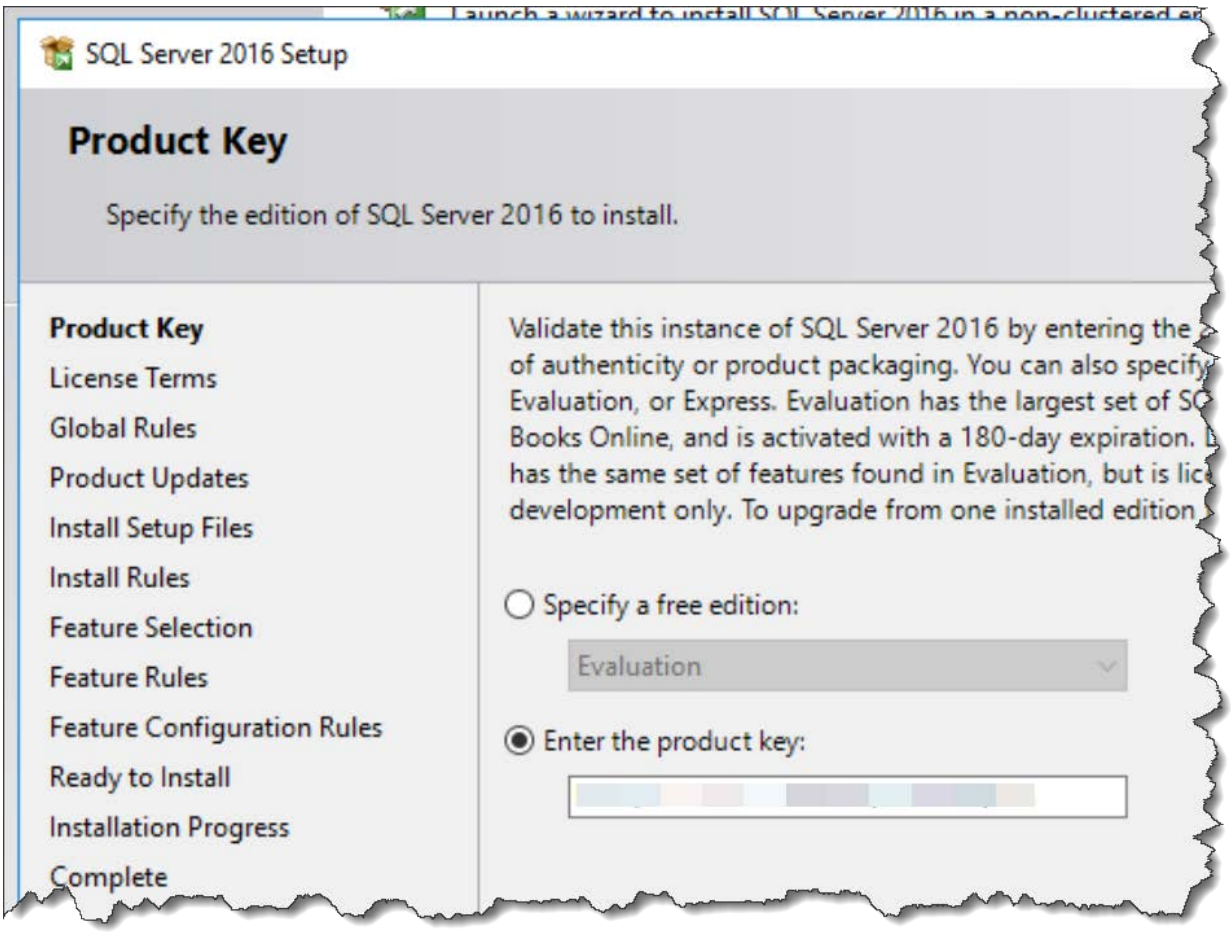
- Run the SQL Server setup program
- Click **Yes** on any User Account Control dialogs that appear

You should now see the SQL Server Installation Center window.

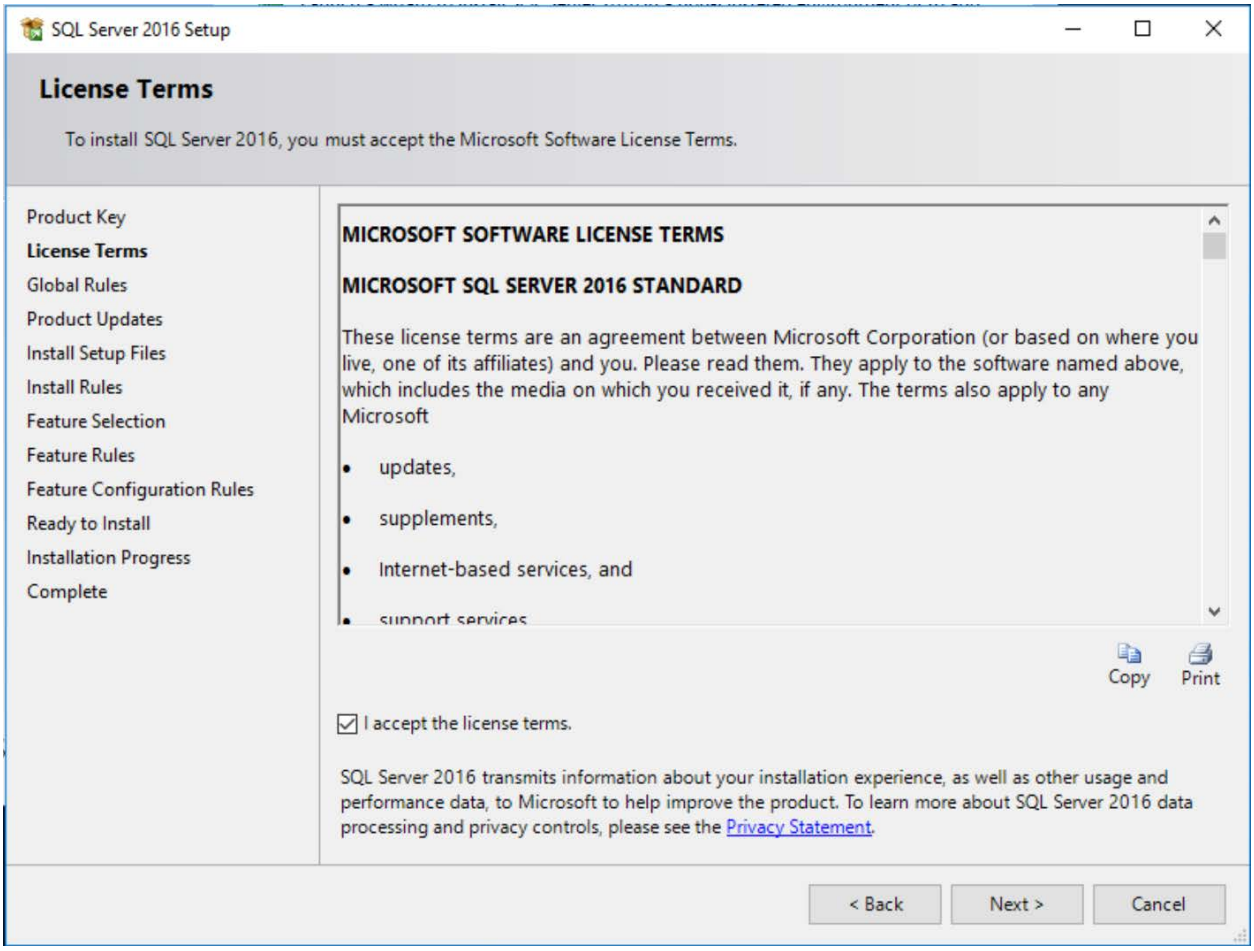


- In the left panel, click **Installation**
- In the right panel, click **New SQL Server stand-alone installation or add features to an existing installation**

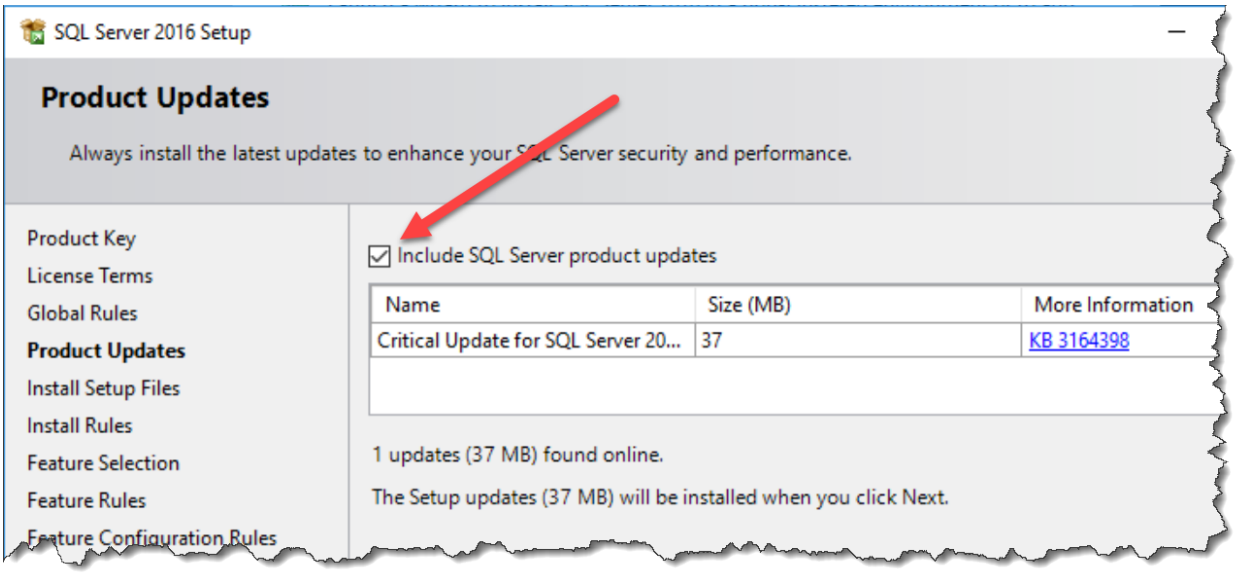
You might now be prompted for a license key.



- Enter a product key
- Click **Next**

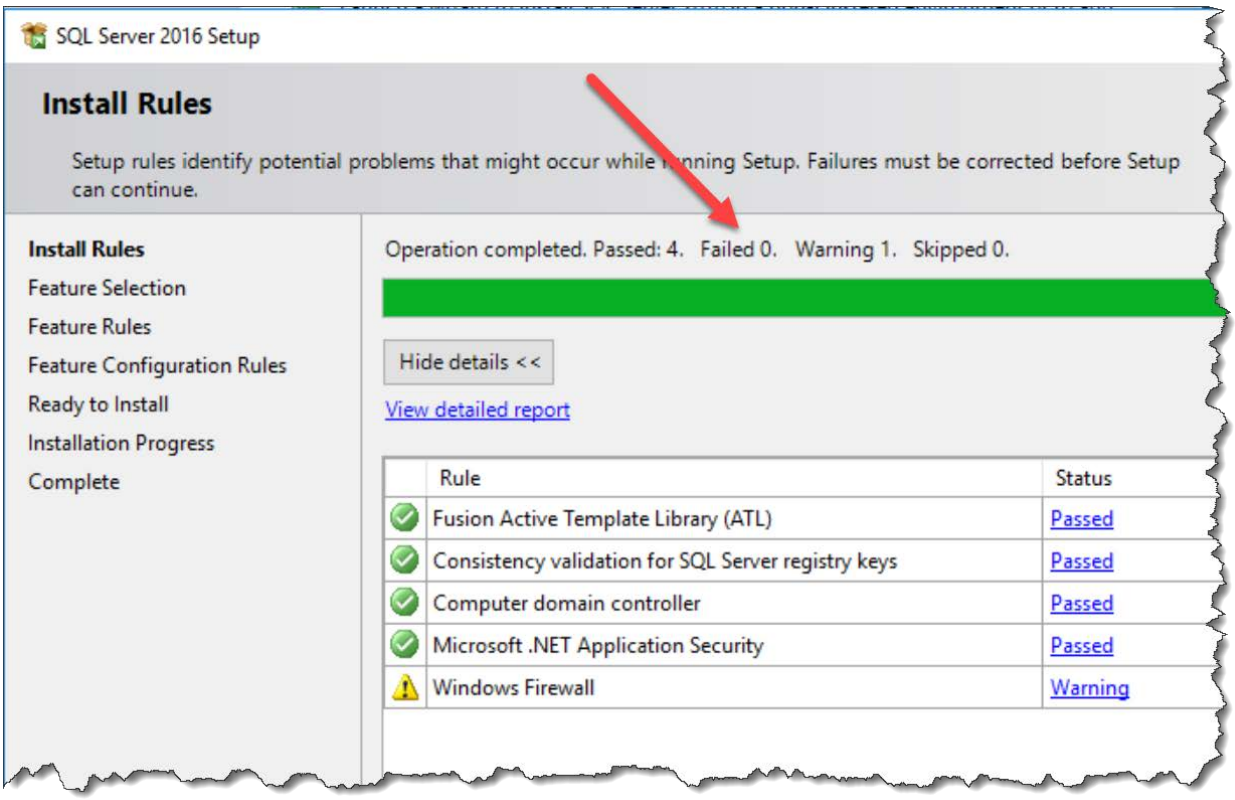


- Check **I accept the license terms**
- Click **Next**



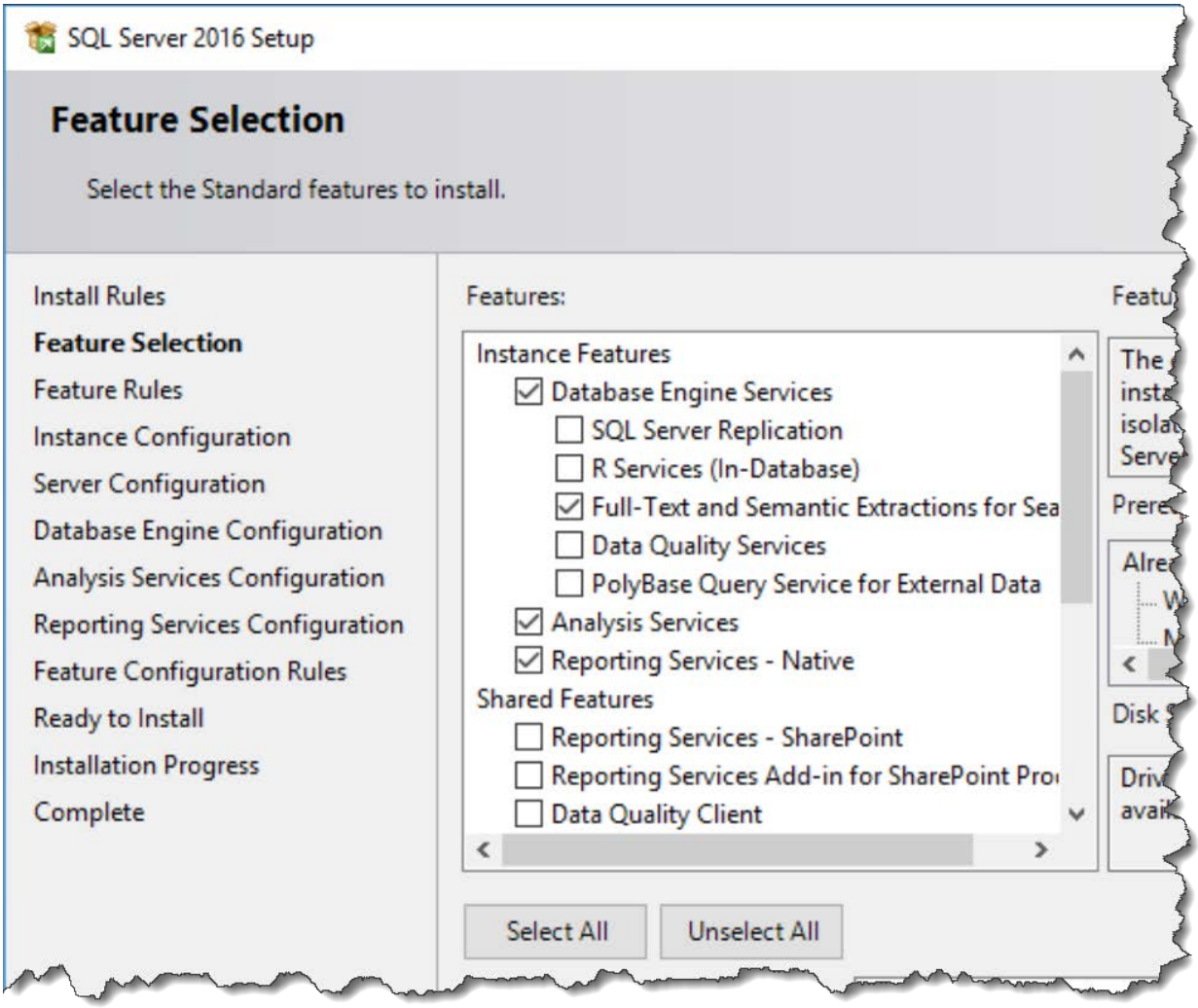
- Check **Include SQL Server product updates**
- Click **Next**

Verify that none of the install rule checks have failed.



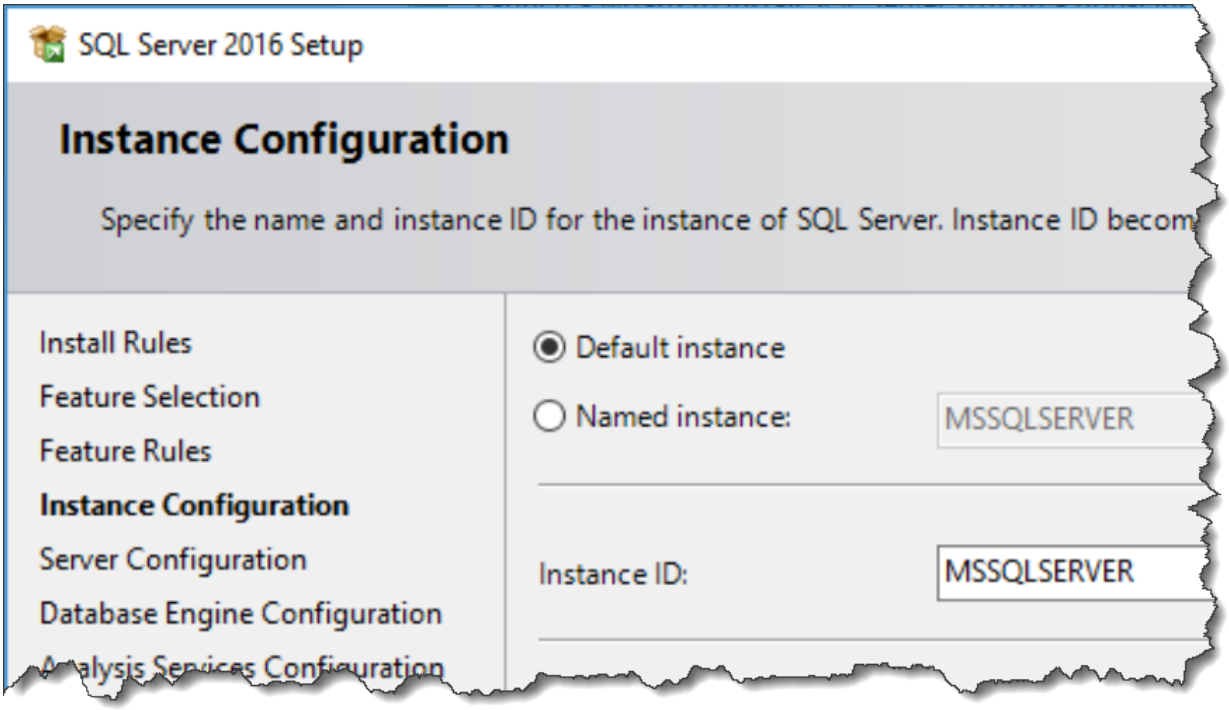
- Verify that there are 0 failures
- Click **Next**

You should now be on the **Feature Selection** page.



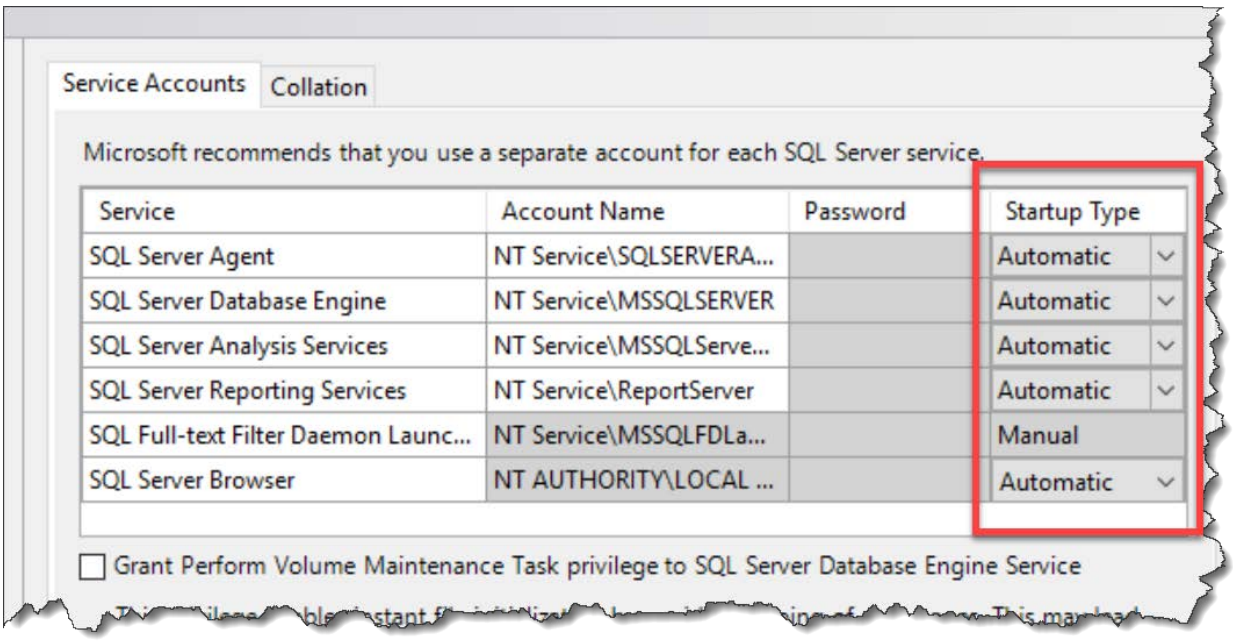
- Under **Instance Features** check
  - **Database Engine Services**
  - **Full-text and Semantic Extractions for Search**
  - **Analysis Services**
  - **Reporting Services – Native**
- Click **Next**

You should now be on the Instance Configuration page.

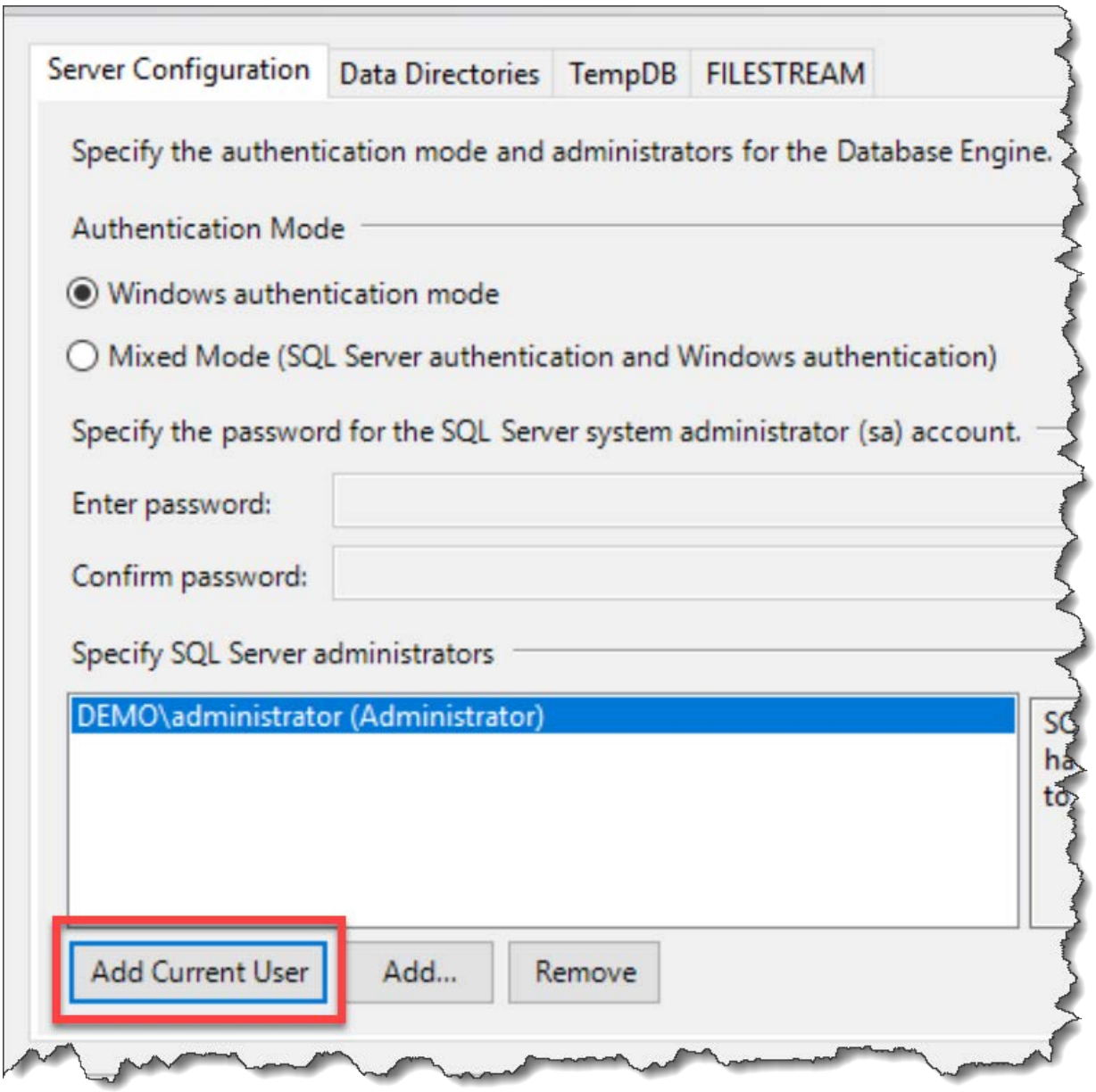


- Choose **Default instance**
- Click **Next**

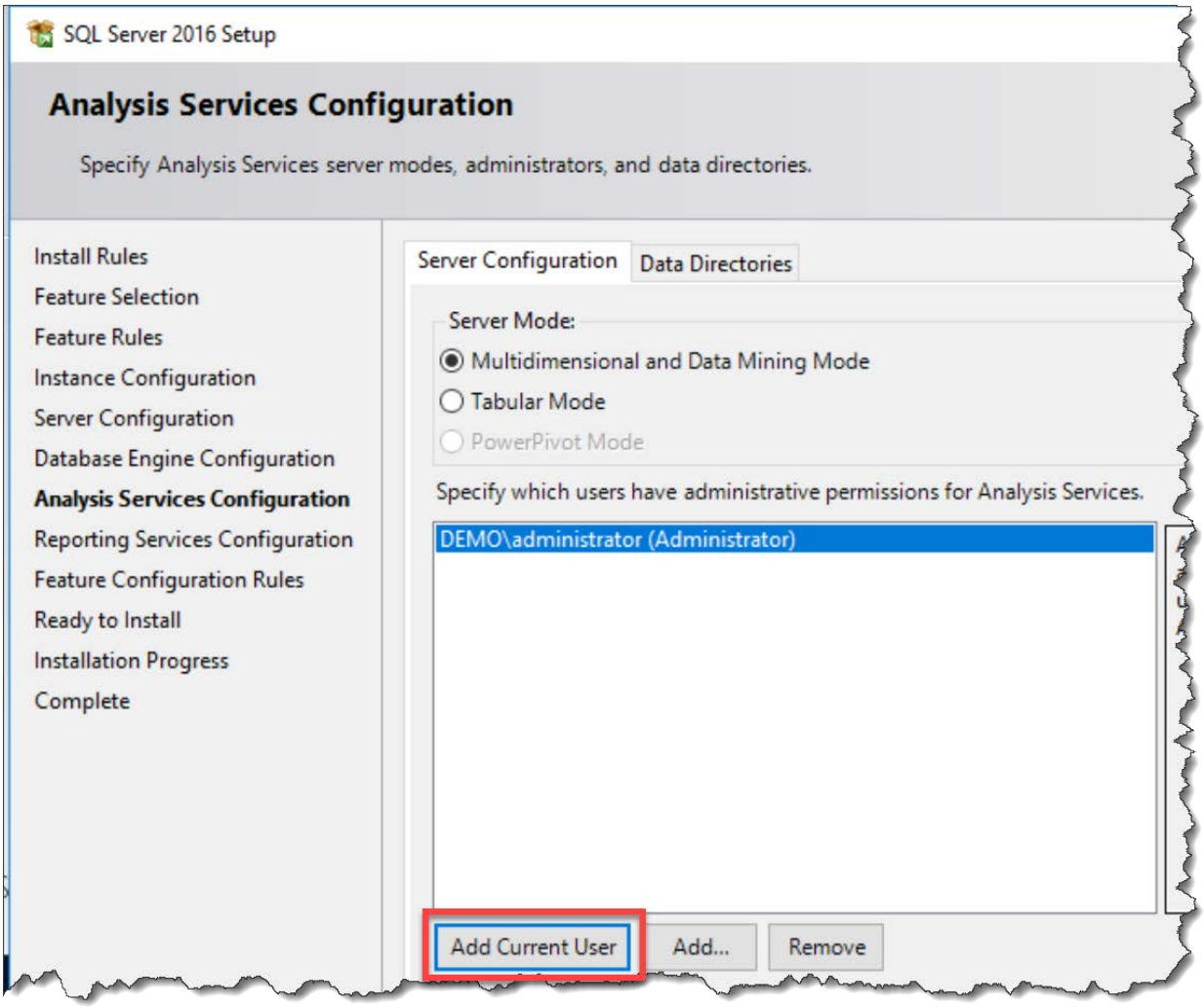
On the **Server Configuration** page you need to set all the services to automatically start.



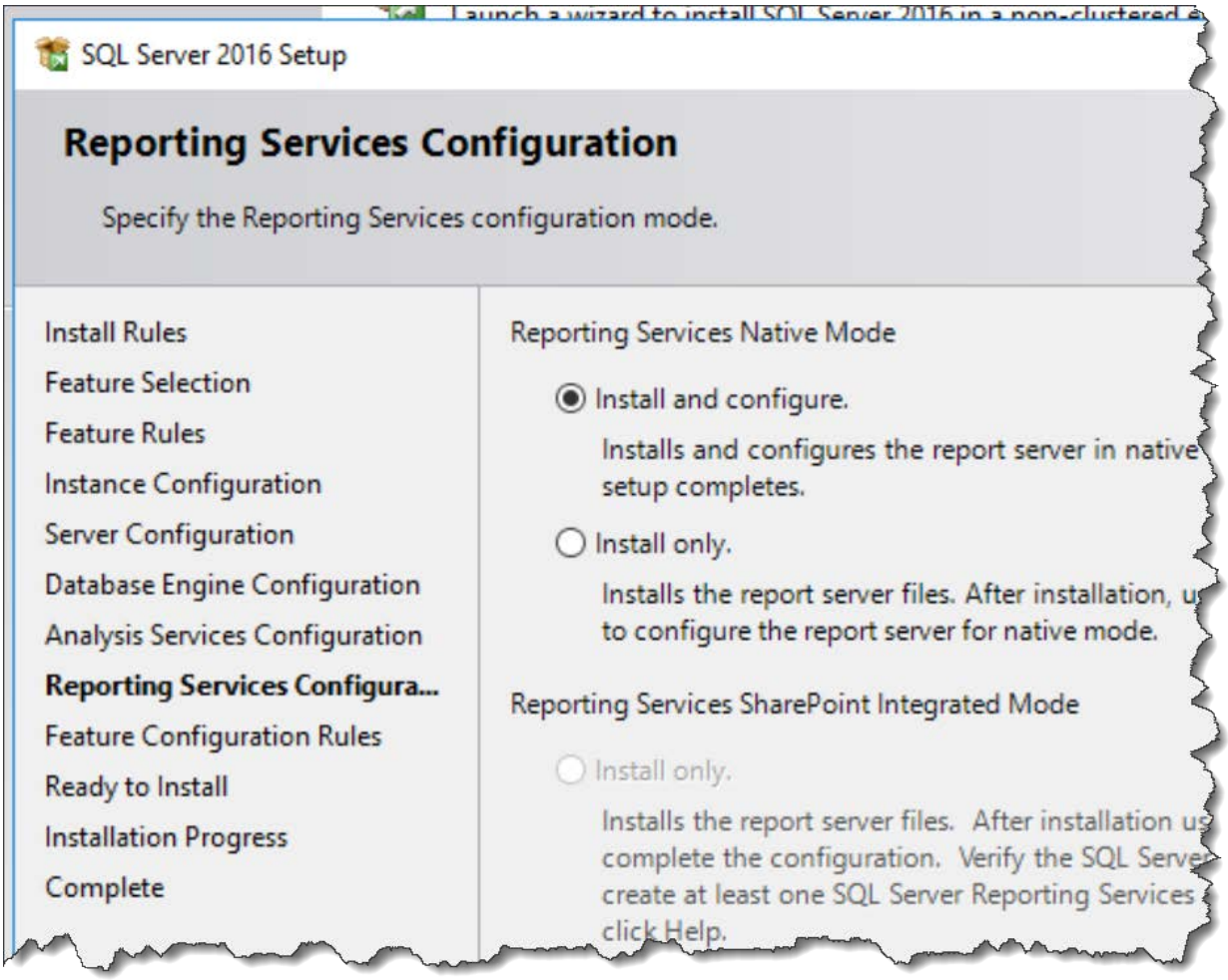
- For each service, set **Startup Type** to **Automatic**
- Click **Next**



- Choose **Windows authentication mode**
- Click the **Add Current User** button to add the current user as a SQL Server administrator
- (Optional) Click the **Add…** button and add the **Domain Admins** group to the SQL Server administrators
- Click **Next**

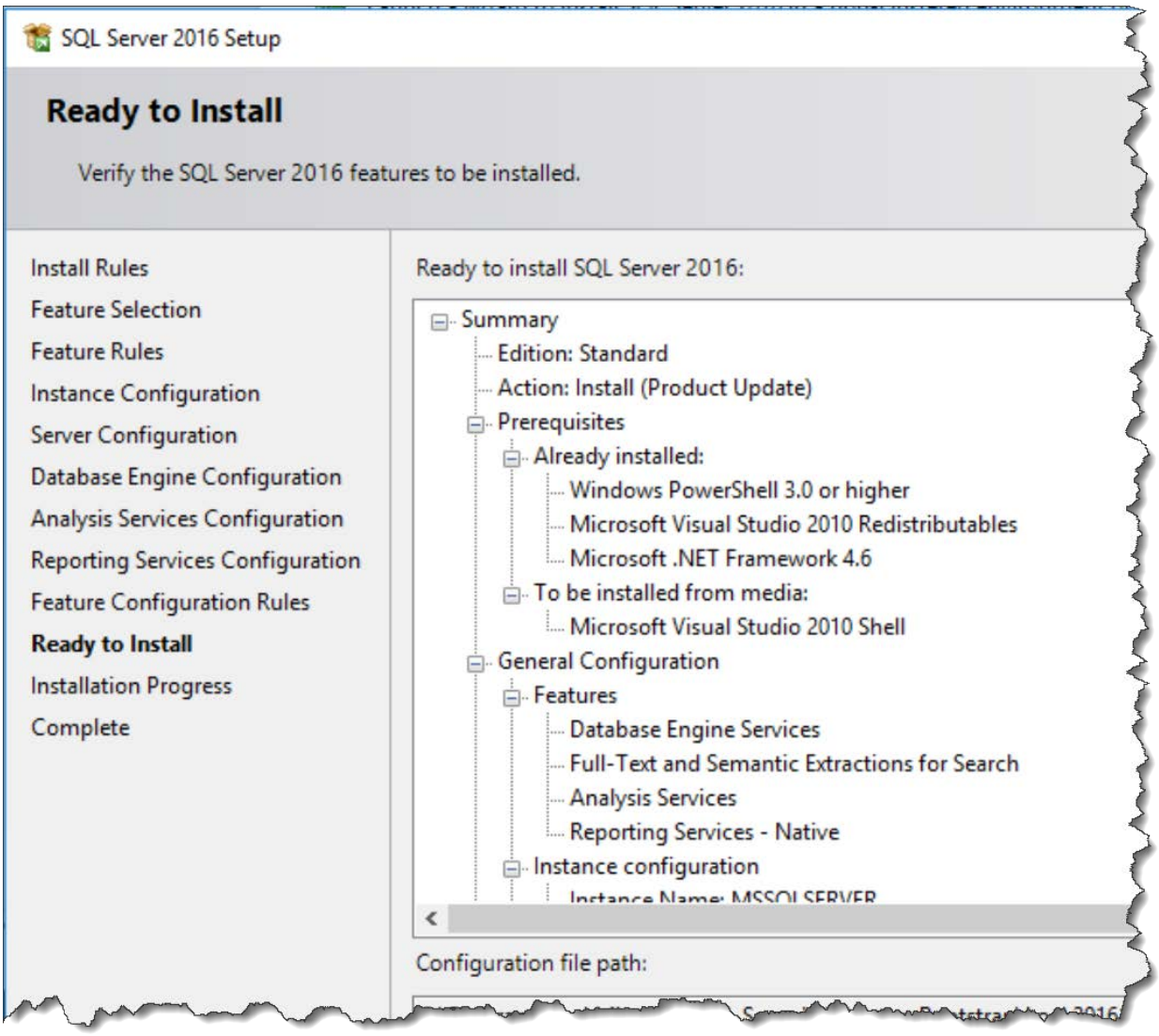


- Click the **Add Current User** button to add the current user as an Analysis Server administrator
- (Optional) Click the **Add…** button and add the **Domain Admins** group to the Analysis Services administrators
- Click **Next**



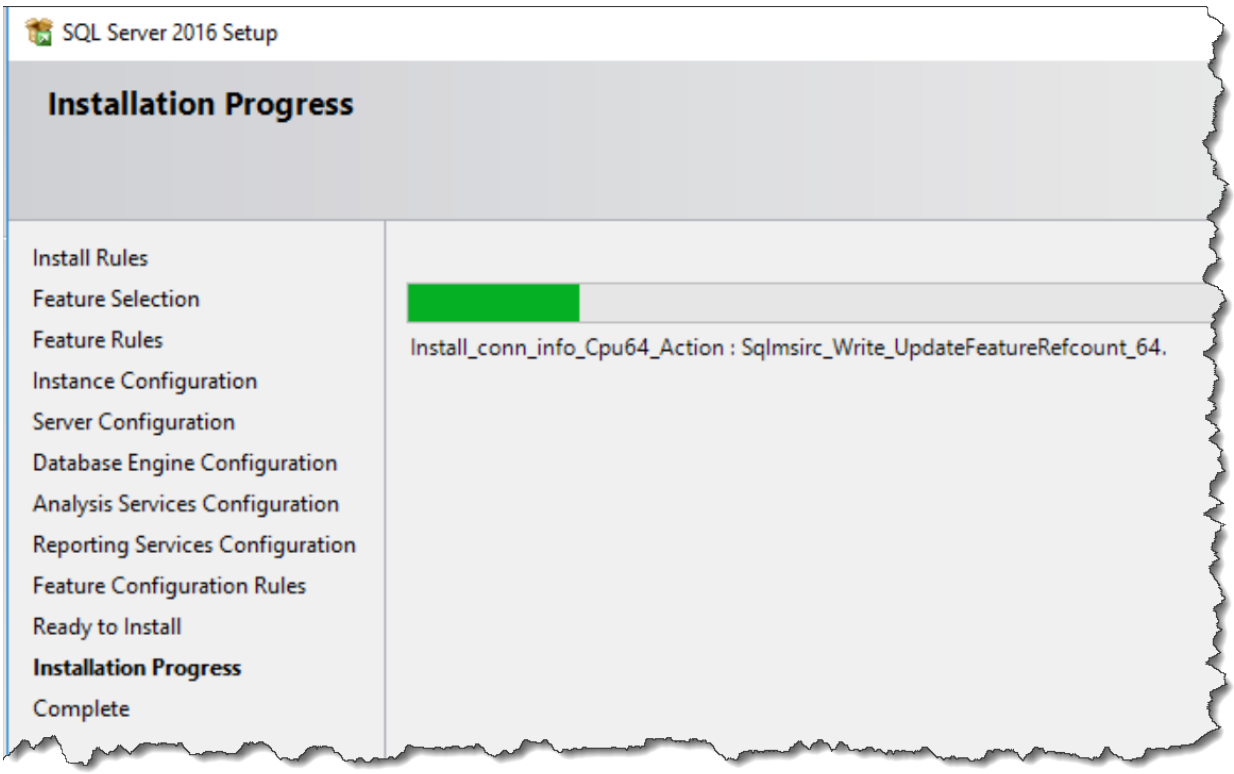
- Choose **Install and configure**
- Click **Next**

You should now see the **Ready to Install** page.

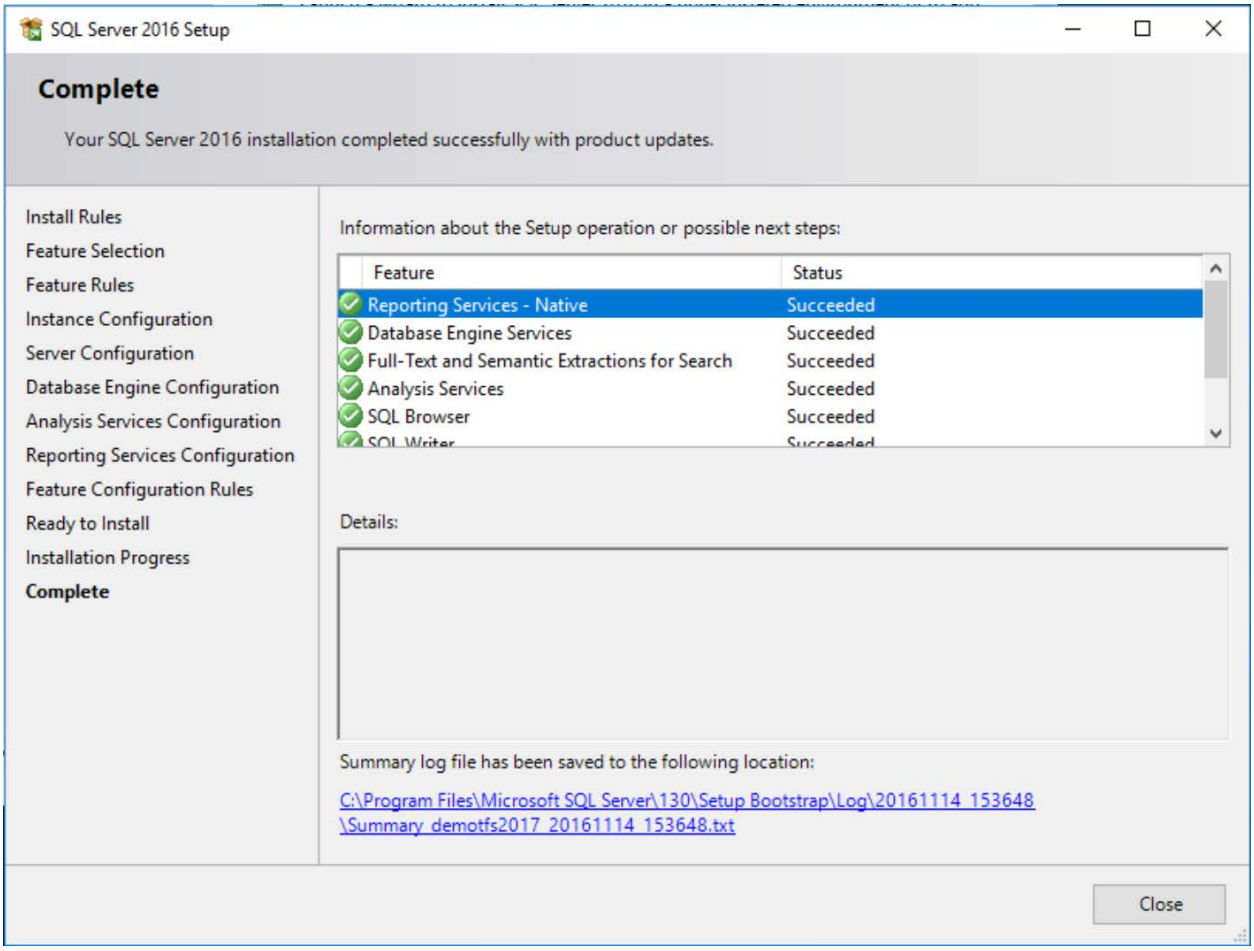


- Click **Install**

The installer should now be running.

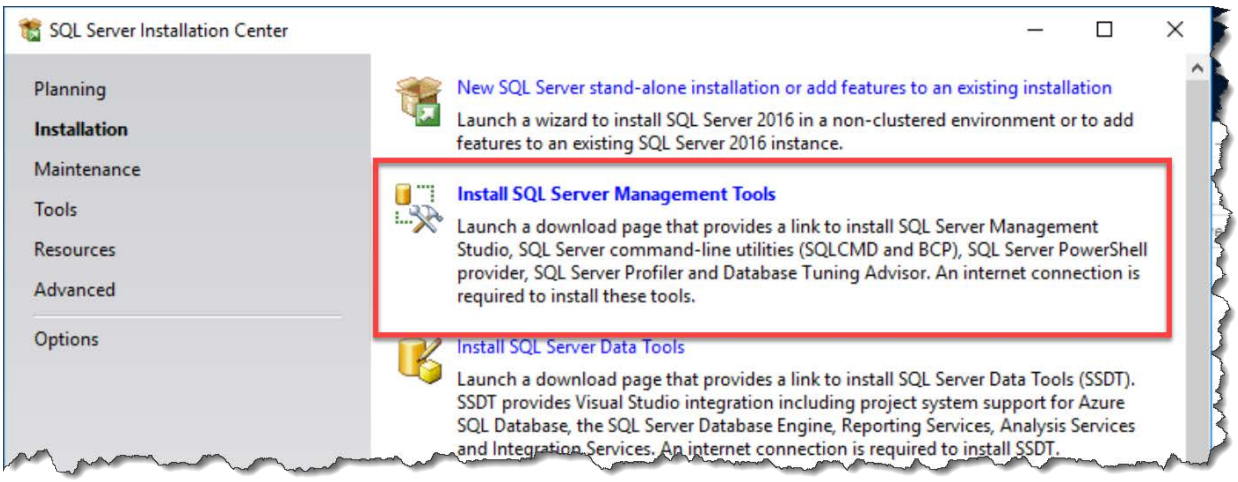


Eventually, the installer should finish.



- Verify that all items installed successfully
- Click **Close** to exit the installer

You should now be back on the **SQL Server Installation Center**.



- (Optional) Install **SQL Server Management** Tools
- Or click the close button to exit the installer

SQL Server 2016 is now installed.

- (Recommended) Re-run **Windows Update** and install any available updates

Chapter 4: Install Team Foundation Server 2017

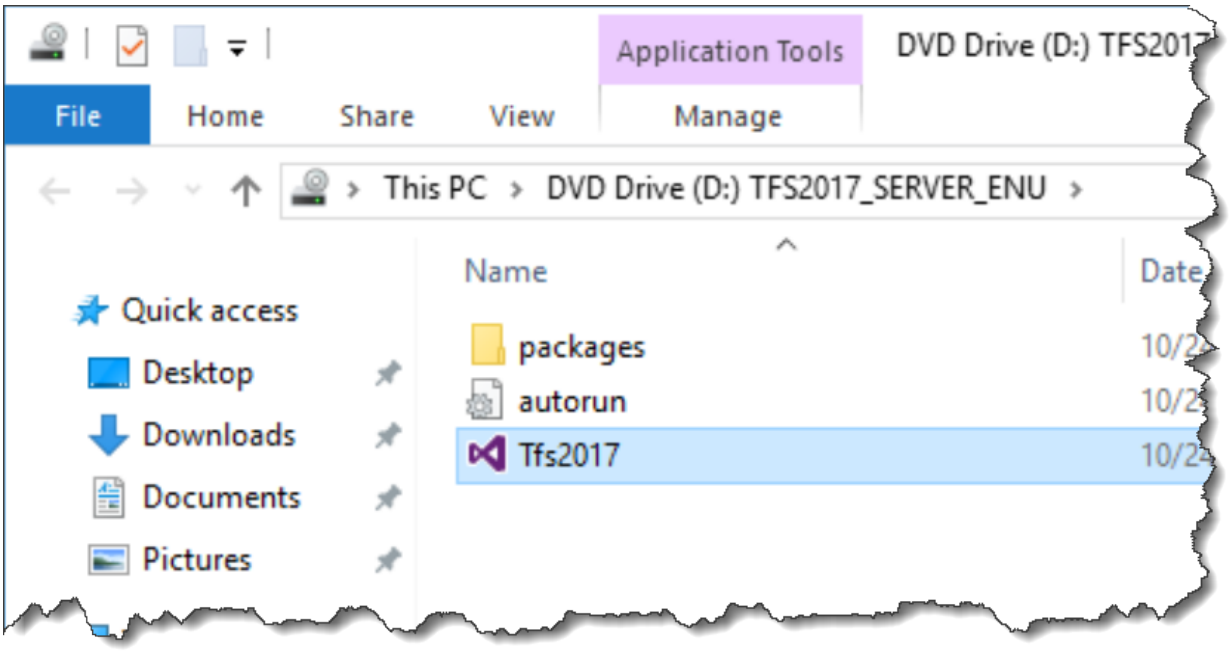
Introduction

Now that Windows and SQL Server are installed, you're ready to install Team Foundation Server 2017 (TFS2017). You'll probably want to create three domain accounts for use by the various pieces of TFS: TFS Service (*domain*\tfsservice), TFS Reports (*domain*\tfsreports), and TFS Build (*domain*\tfsbuild).

- If you're installing this on a Hyper-V virtual machine with dynamic memory enabled, change the **minimum amount of RAM to 2GB** (at least temporarily) to allow Team Foundation Server 2017 to install along with SQL Server.
- Gather the username and passwords for the 3 TFS service accounts (see above)
- Log on to the server using a user account with Administrator privileges

Run the Installer

- Mount the TFS2017 ISO image or insert a TFS2017 installer DVD
- Using Windows Explorer (explorer.exe), navigate to the installer directory

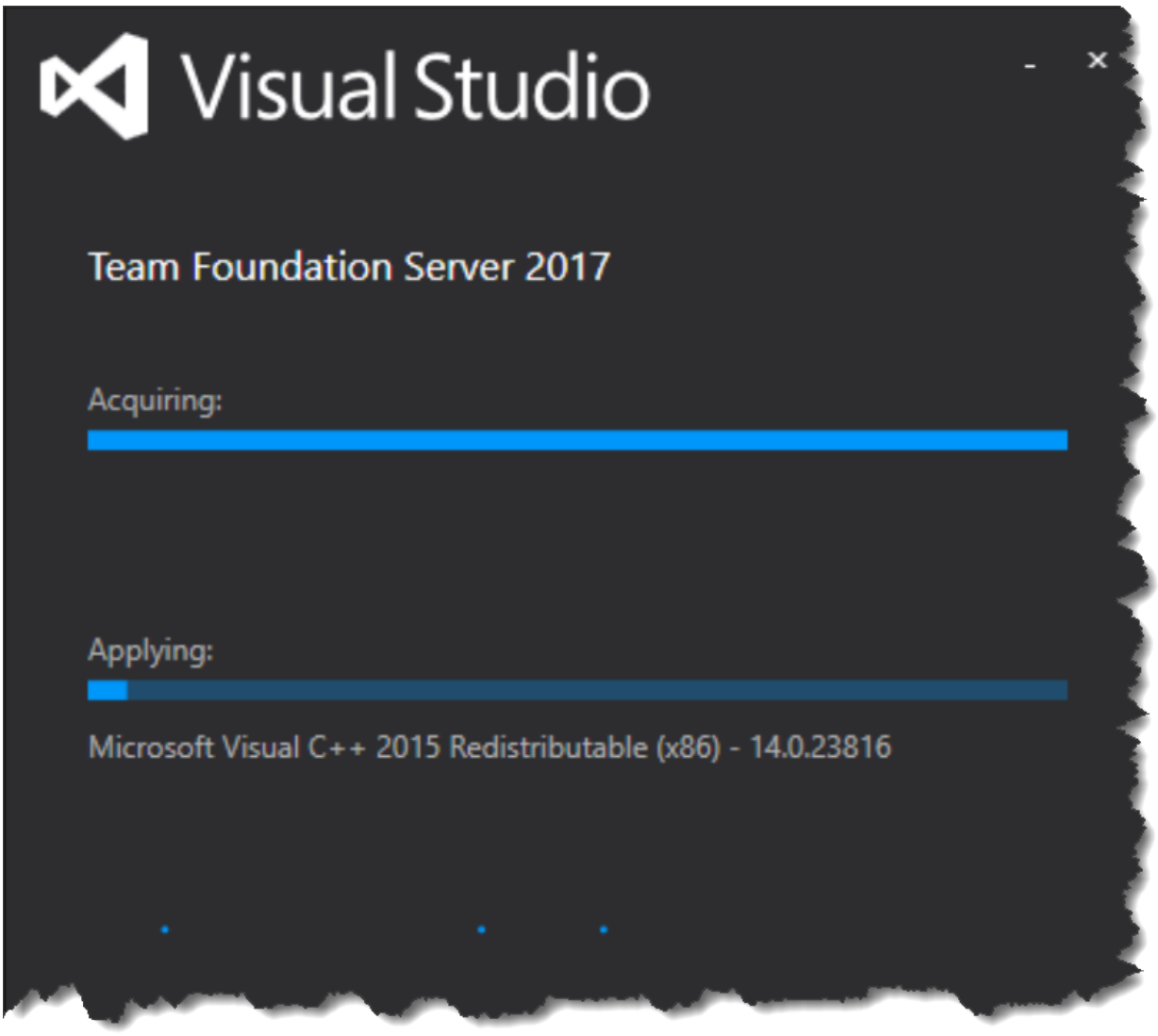


- Run **Tfs2017.exe**

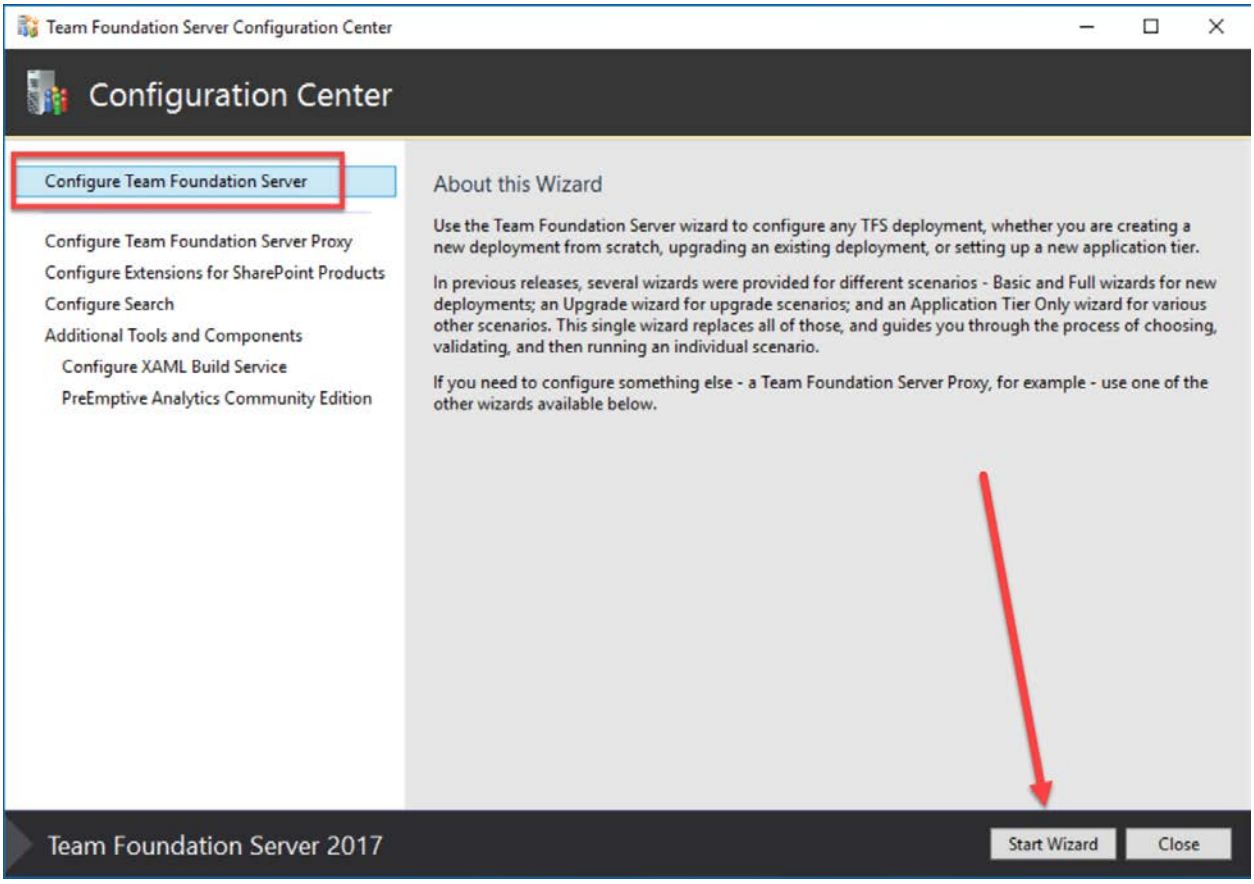
You should see the **Team Foundation Server Setup** dialog.

- Click **Install**

The installer will run and start to copy files to your disk.

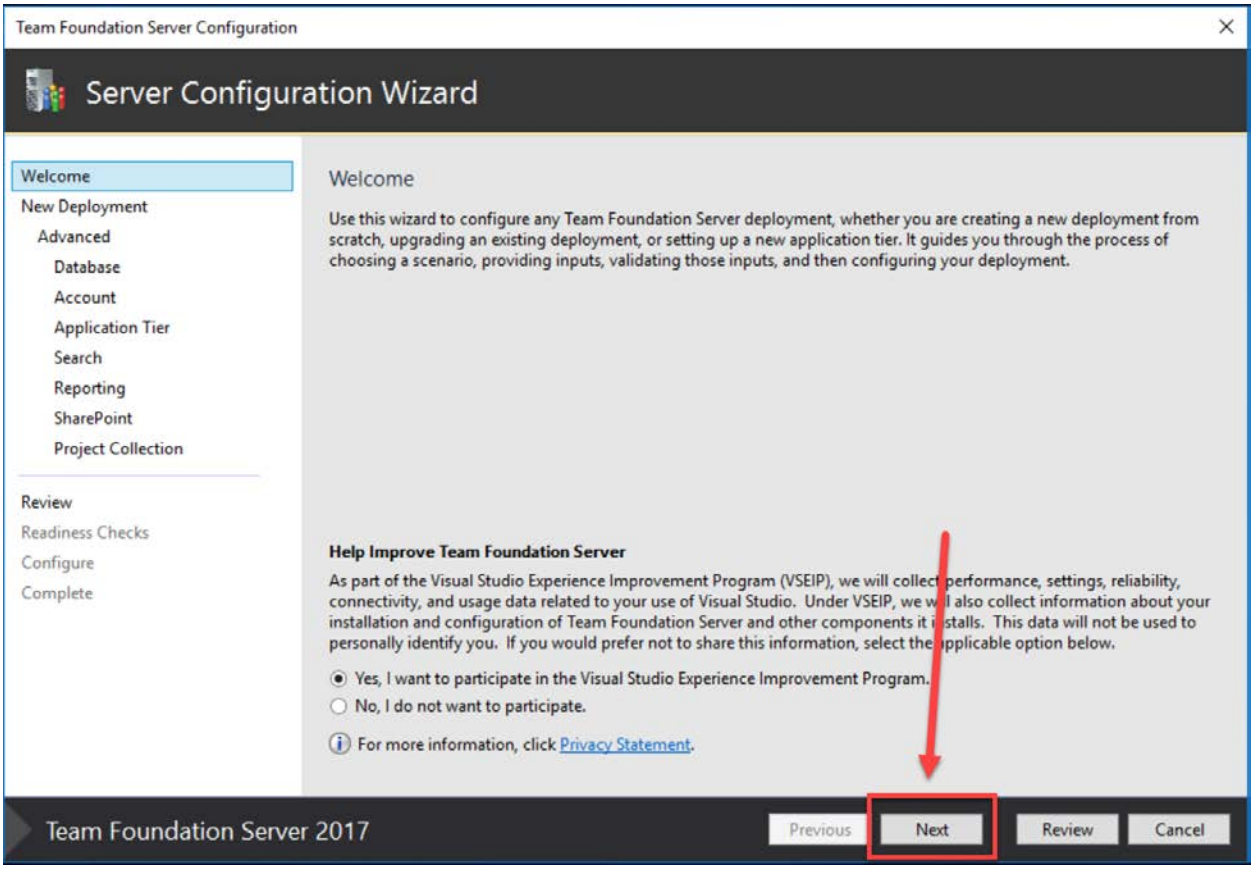


You should now see the **Team Foundation Server Configuration Center**.



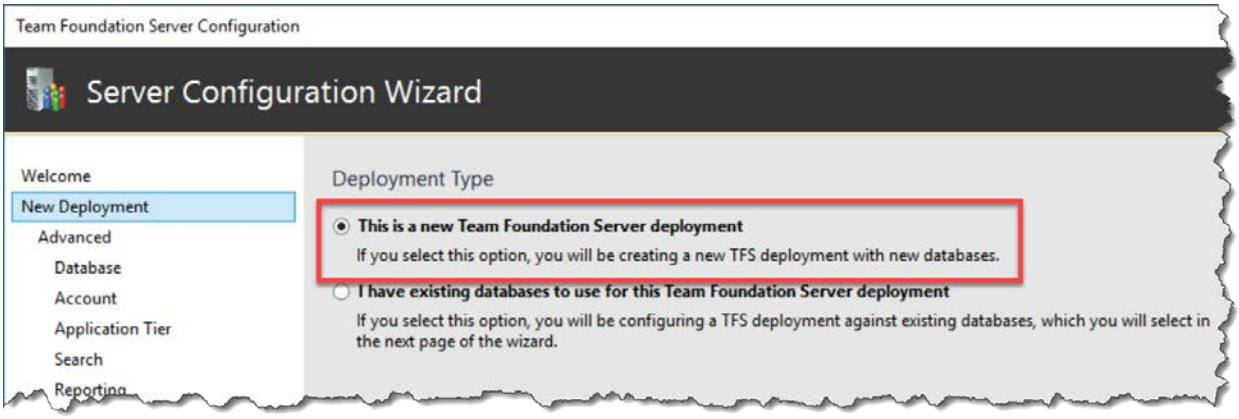
- Choose **Configure Team Foundation Server**
- Click **Start Wizard**

You should now be on the welcome page.



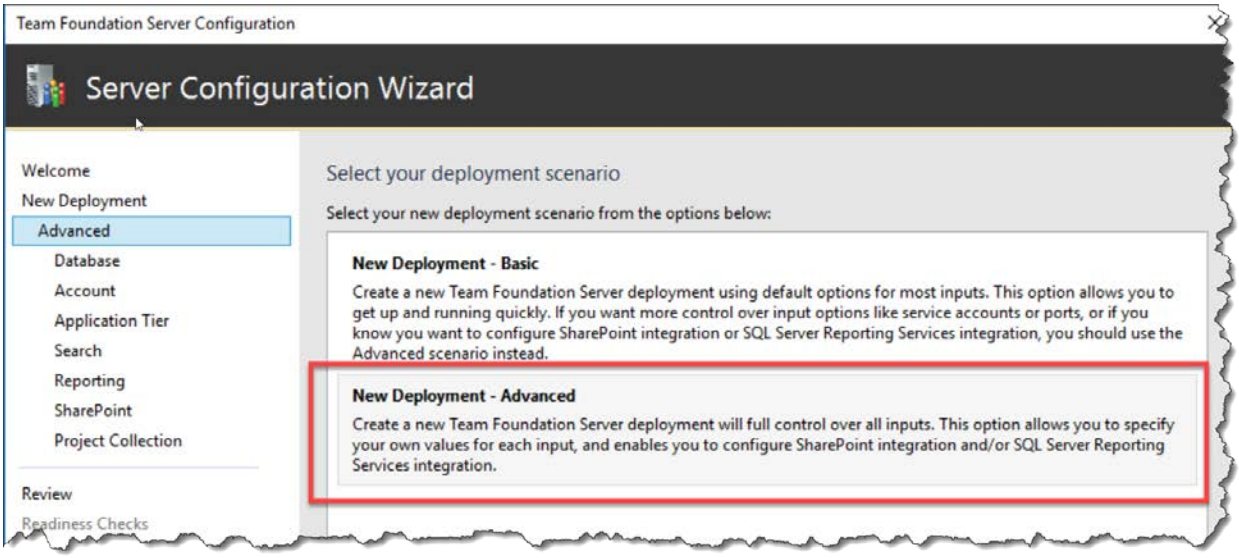
- Choose **Yes, I want to participate in the Visual Studio Experience Improvement Program**
- Click **Next**

You should now be on the Deployment Type page.



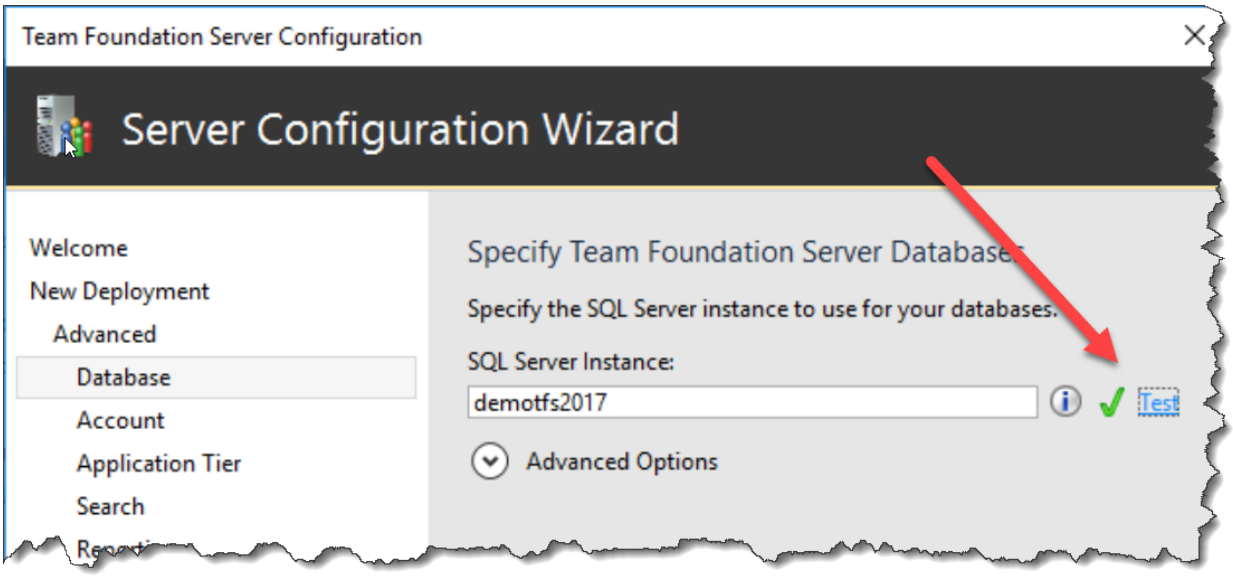
- Select **This is a new Team Foundation Server deployment**
- Click **Next**

On the **Select your deployment scenario** page, you'll be prompted to choose between a TFS Basic or TFS Advanced deployment. Since this guide assumes that you're using SQL Server Reporting Services with TFS, you'll need to choose the Advanced version.



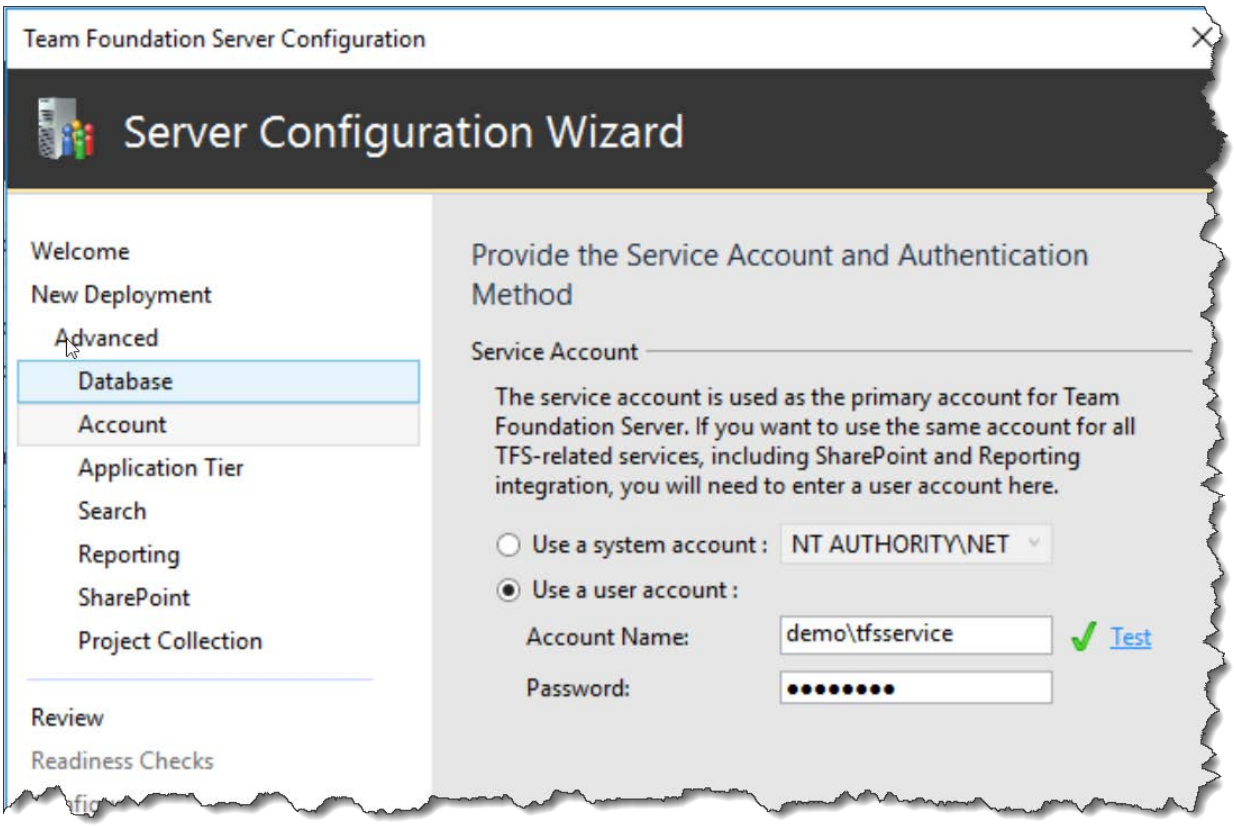
- Choose **New Deployment – Advanced**
- Click **Next**

The **Specify Team Foundation Server Databases** page prompts you to choose your SQL Server database. This guide assumes that you installed SQL Server 2016 Standard on the same machine as TFS.



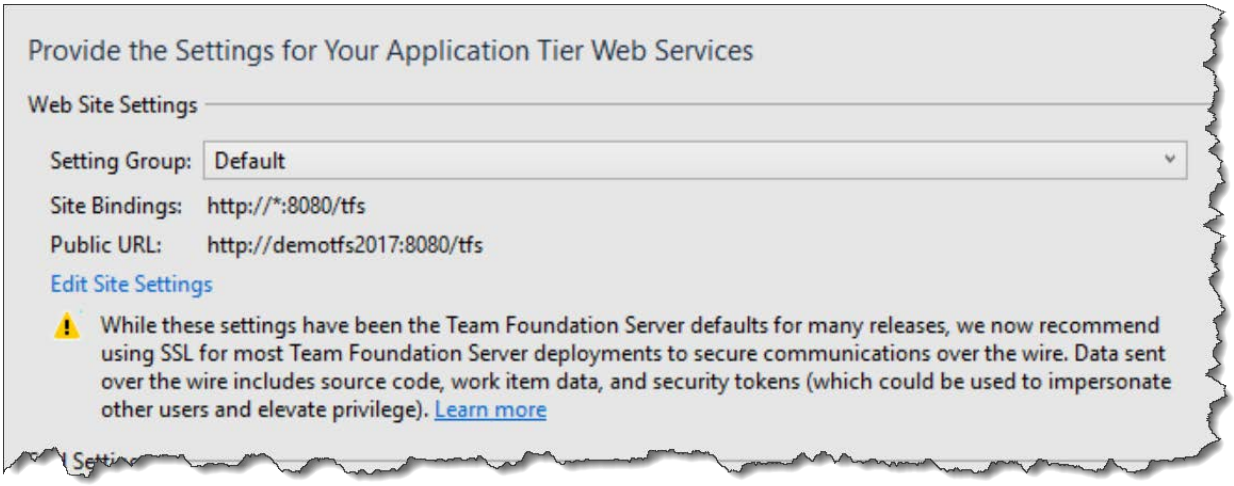
- To the right of the **SQL Server Instance** textbox, click the **Test** link to verify the connection to SQL Server
- Confirm that the test passes
- Click **Next**

You should now see the Service Account page. You can choose to run TFS as a system account but I find that this makes permissions management – more specifically, permissions *debugging* – much harder later on when you start doing automated builds, automated deployments, and automated testing. My recommendation is to run TFS as a separate service account.

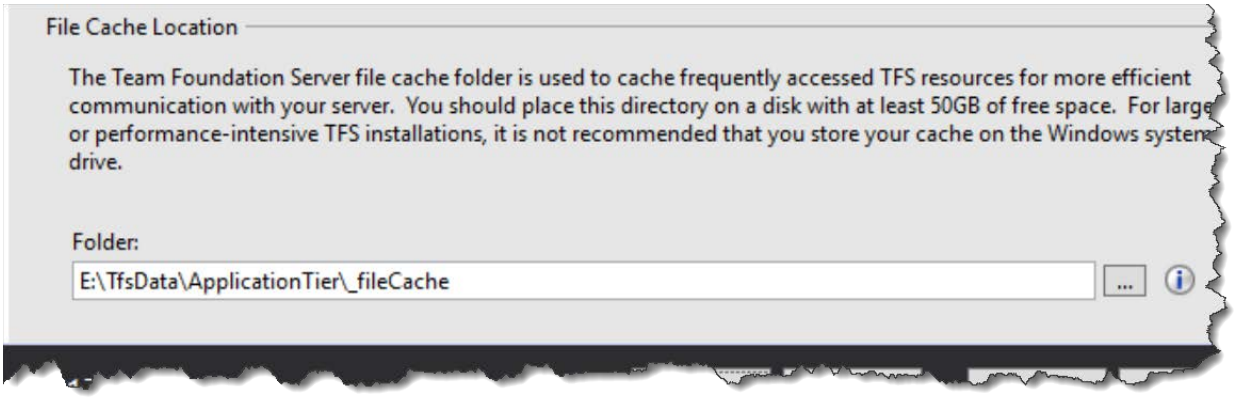


- Choose **Use a user account**
- In the **Account Name** textbox, type the fully-qualified name of the service account. Example: demo\tfsservice
- In the **Password** textbox, enter the password for the service account
- Click the **Test** link to verify the credentials are correct
- Click **Next**

You should now see a page prompting you for the configuration of TFS in IIS. You may see a warning about using SSL encryption. It's a good idea but it's not required.



(Optional) At the bottom of this page, there's a section for **File Cache Location**. TFS caches files for efficiency. The contents of this directory can become impressively large. For performance reasons and for disk space management reasons, you probably should put this on a separate disk – ideally on a different "spindle" – than your system/operating system drive.



- (Optional) Change the **Folder** path to reference the desired location and disk.
- Click **Next**

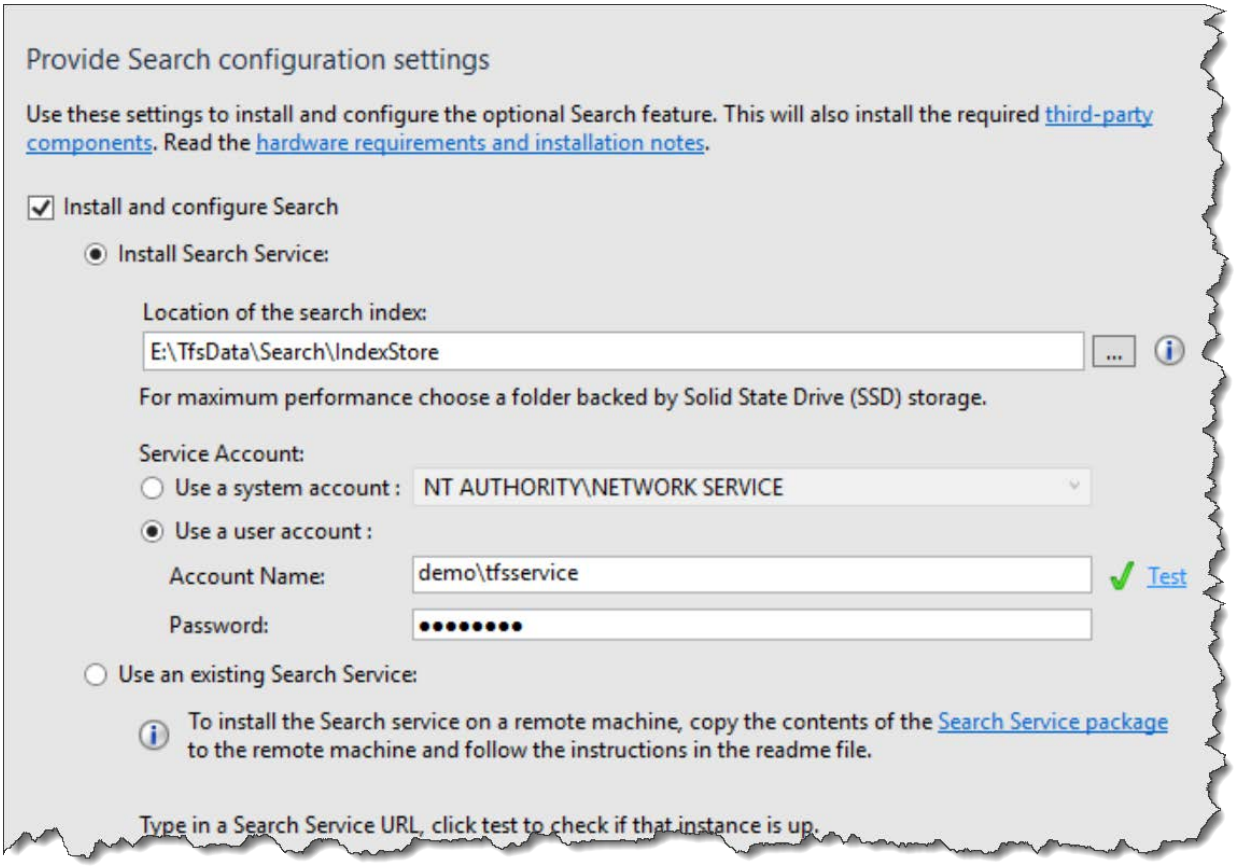
TFS2017 adds a bunch of new features to help you search the contents of your team projects. This is an optional feature.

Option 1: If you *do not* want to install Search:

- Uncheck **Install and configure Search**
- Click **Next**

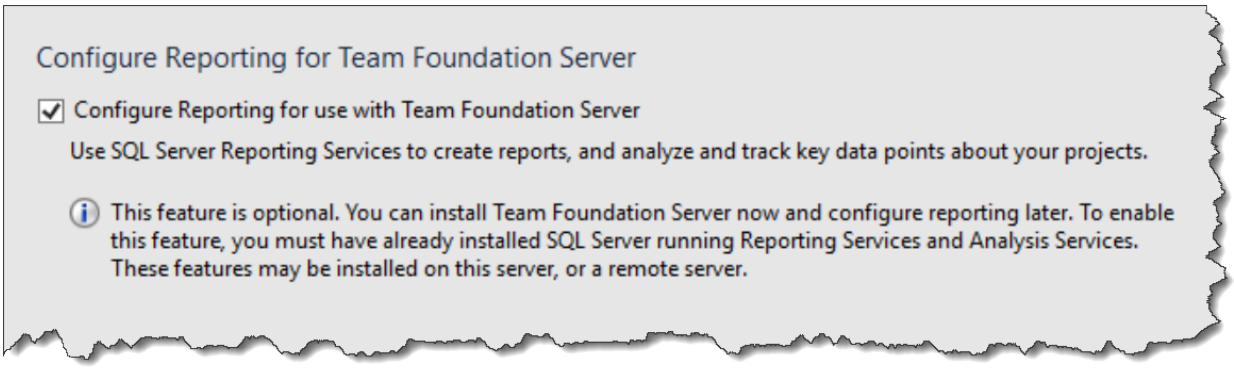
Option #2: Install Search

- Check **Install and configure Search**



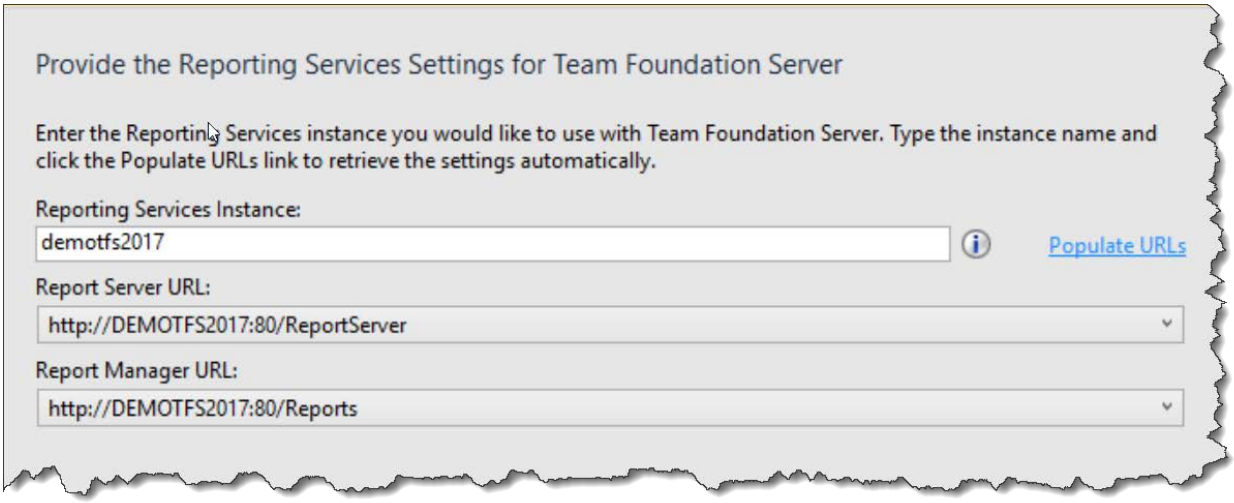
- Choose **Install Search Service**
- Set the **Location of the search index** to the drive and folder you want to use for search. For performance reasons, you'll probably want to keep this on a different drive than the system drive. If your TFS installation is large and busy, you may want to put this on its own drive by itself. In the screenshot, I'm putting this on the same drive as my TFS File Cache at E:\TfsData\Search\IndexStore.
- Under Service Account choose **Use a user account**
- Set **Account Name** to the service account want to use to run search. In this configuration, I'm using the same account at the TFS Service, demo\tfsservice
- Set the **Password** for the service account
- Click the **Test** link to verify the service credentials
- Click **Next**

You should now see the **Configure Reporting for Team Foundation Server** page. This is another optional feature but this guide assumes that you're installing support for SQL Server Reporting Services with TFS2017.



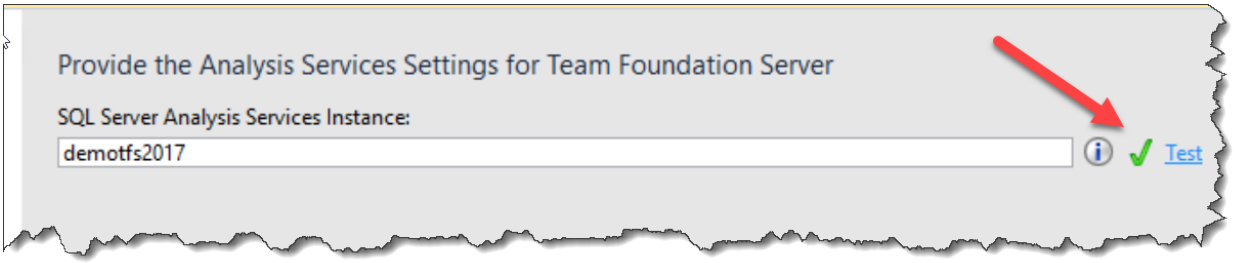
- Check **Configure Reporting for use with Team Foundation Server**
- Click **Next**

You should now see the Provide Reporting Services Settings for Team Foundation Server page. These values should be automatically populated.



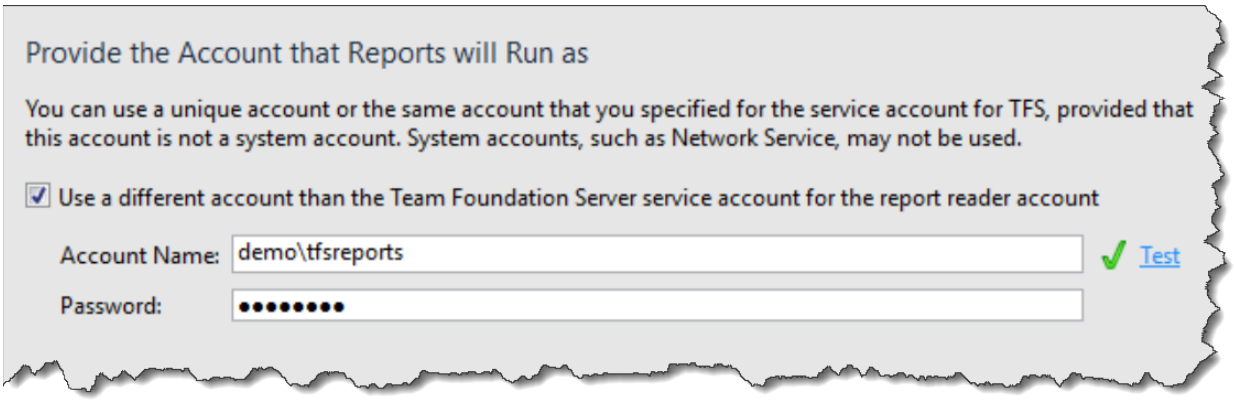
- Click **Next**

You should now see the Analysis Services settings page.



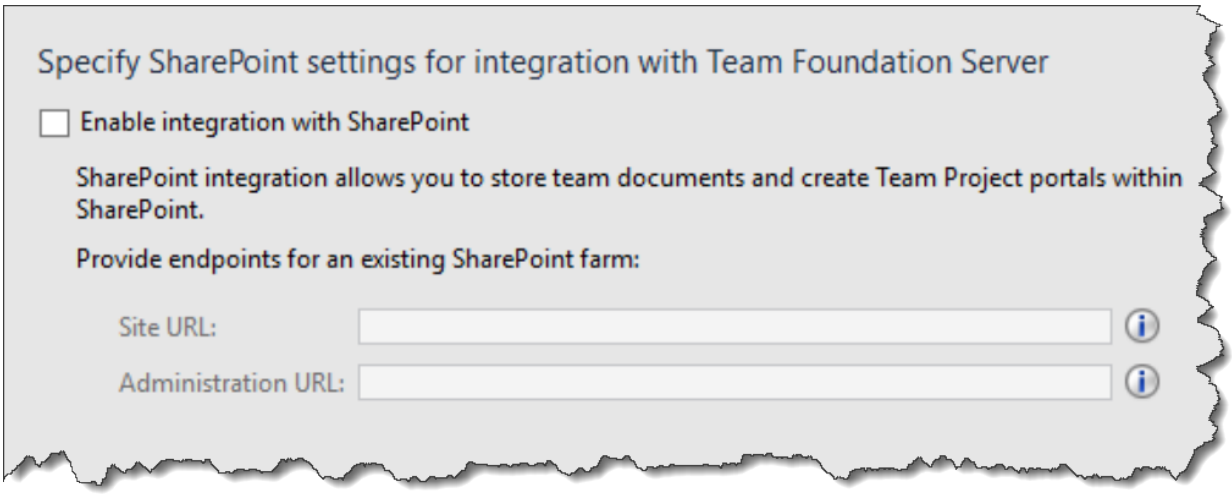
- Click the **Test** link to verify the connection to SQL Server Analysis Services
- Click **Next**

Next you'll provide the credentials for the SQL Server Reporting Services Reports. You have the option of skipping this and running with the same account as the TFS Service Account but this is probably not ideal for security reasons.



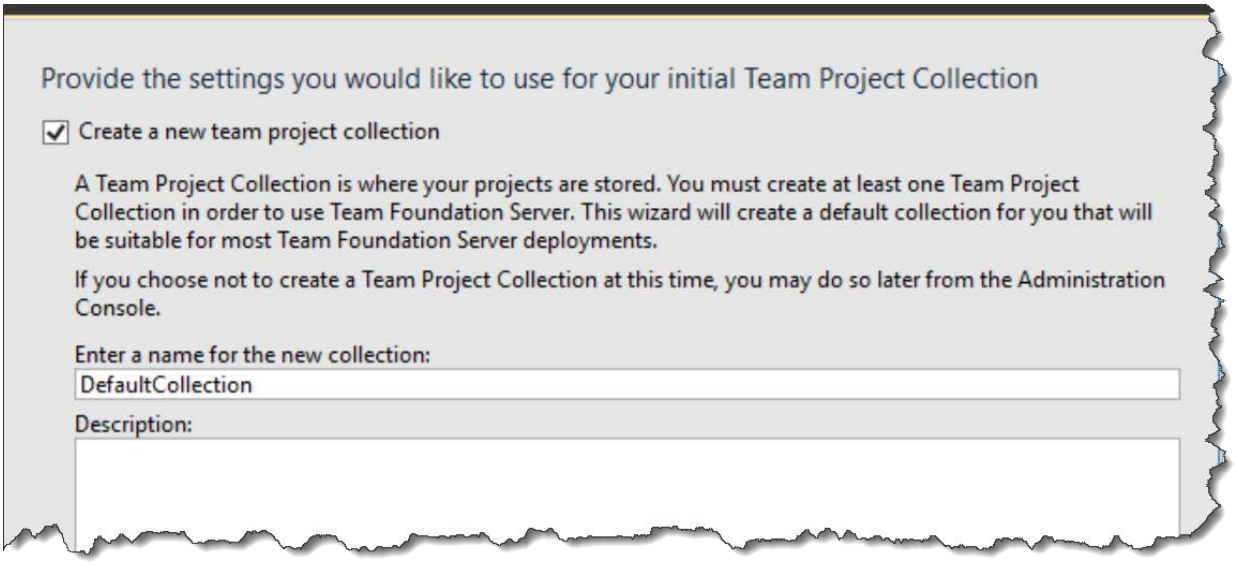
- Check **Use a different account than the Team Foundation Server service account…**
- Set **Account Name** to the fully qualified username of the service account
- Set the **Password** for the service account
- Click the **Test** link to verify the credentials
- Click **Next**

You should now be on the page asking if you'd like to install the integration with SharePoint. The answer is no. No, you have no interest in installing any kind of integration with TFS and SharePoint. THE END.



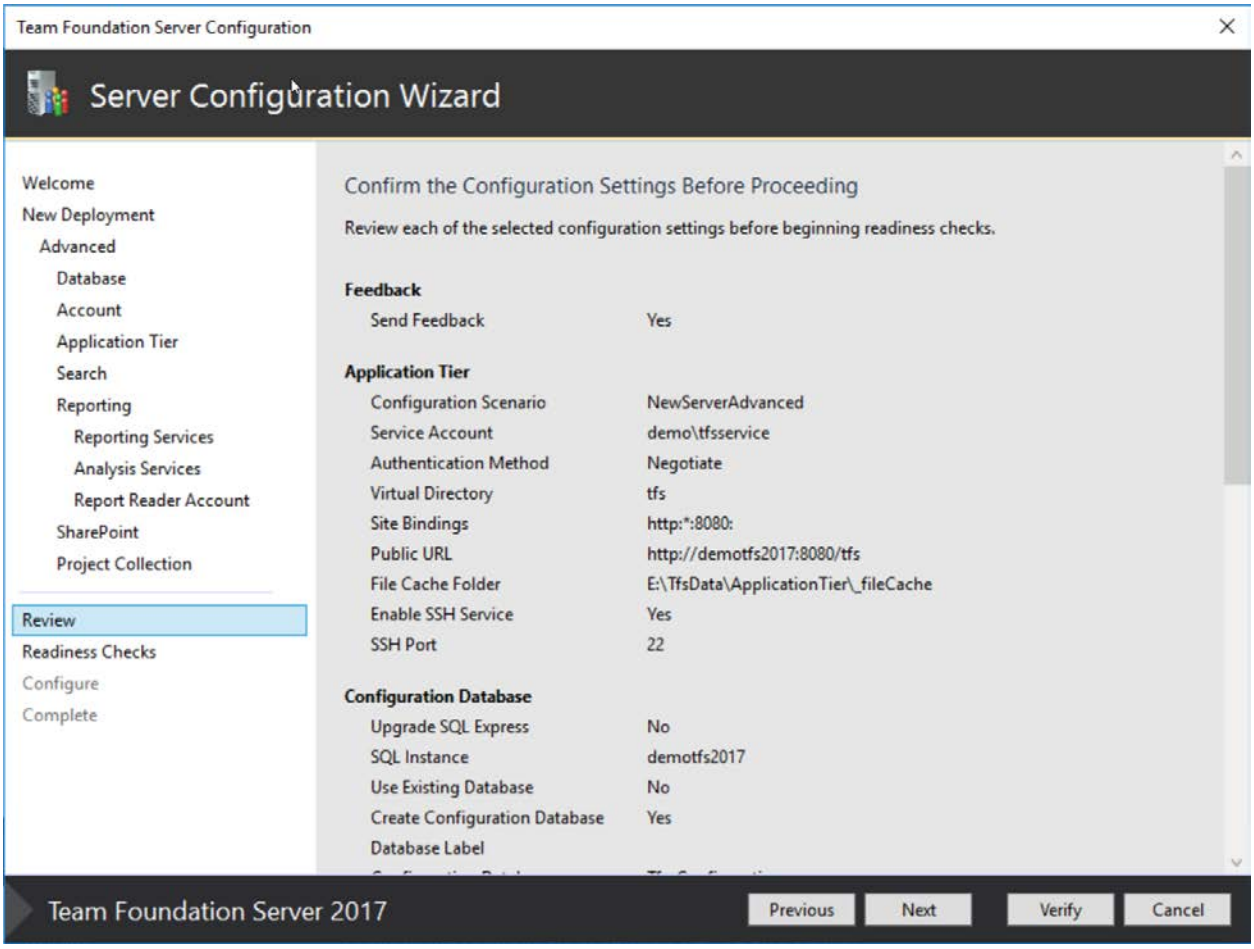
- Uncheck **Enable integration with SharePoint**
- Click **Next**

The installer will now prompt you to create a new Team Project Collection (TPC). The answer to this one (unless you're doing a migration) is yes.



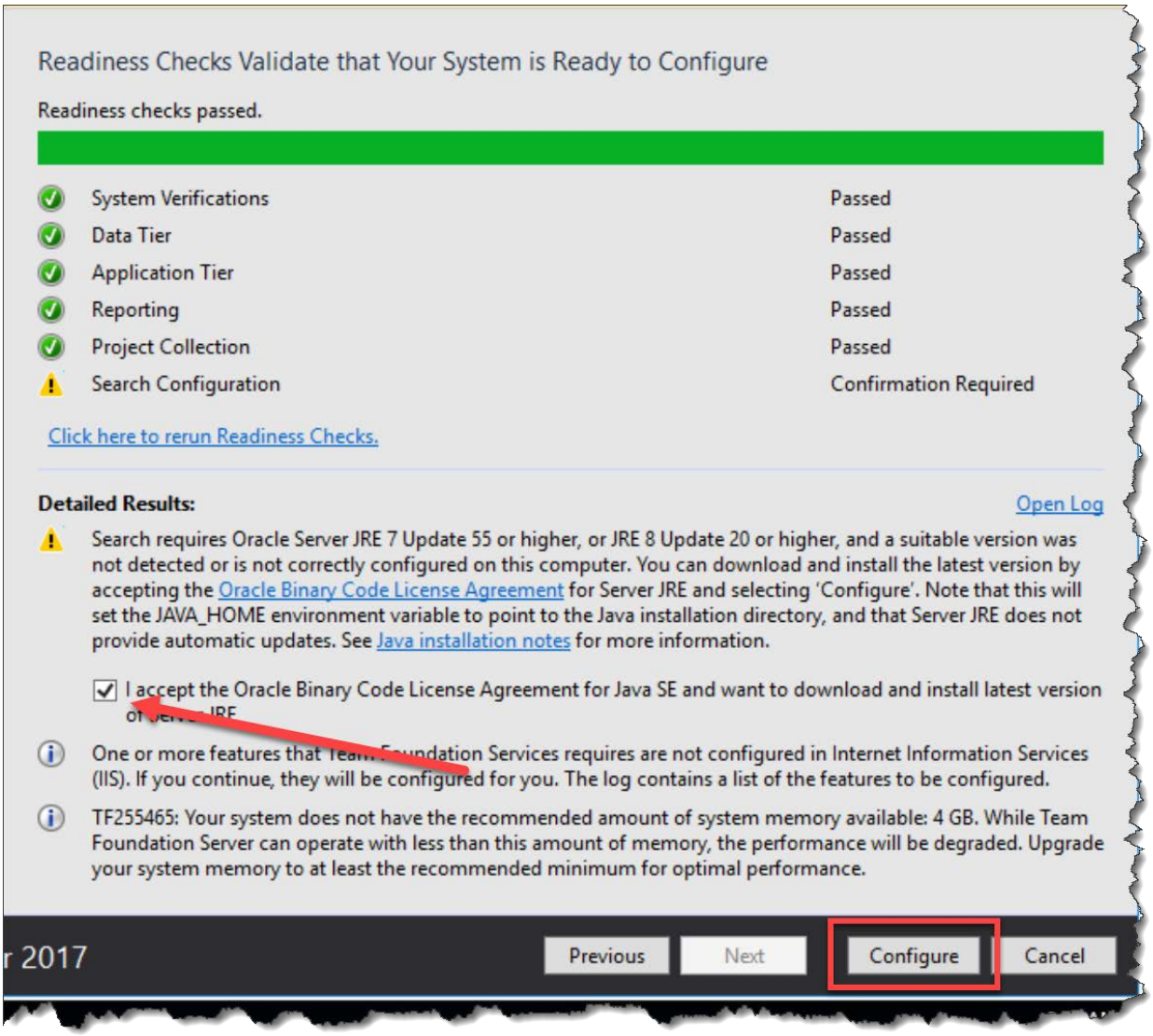
- Check **Create a new team project collection**
- Click **Next**

You should now be on the **Confirm the Configuration Settings Before Proceeding** page.



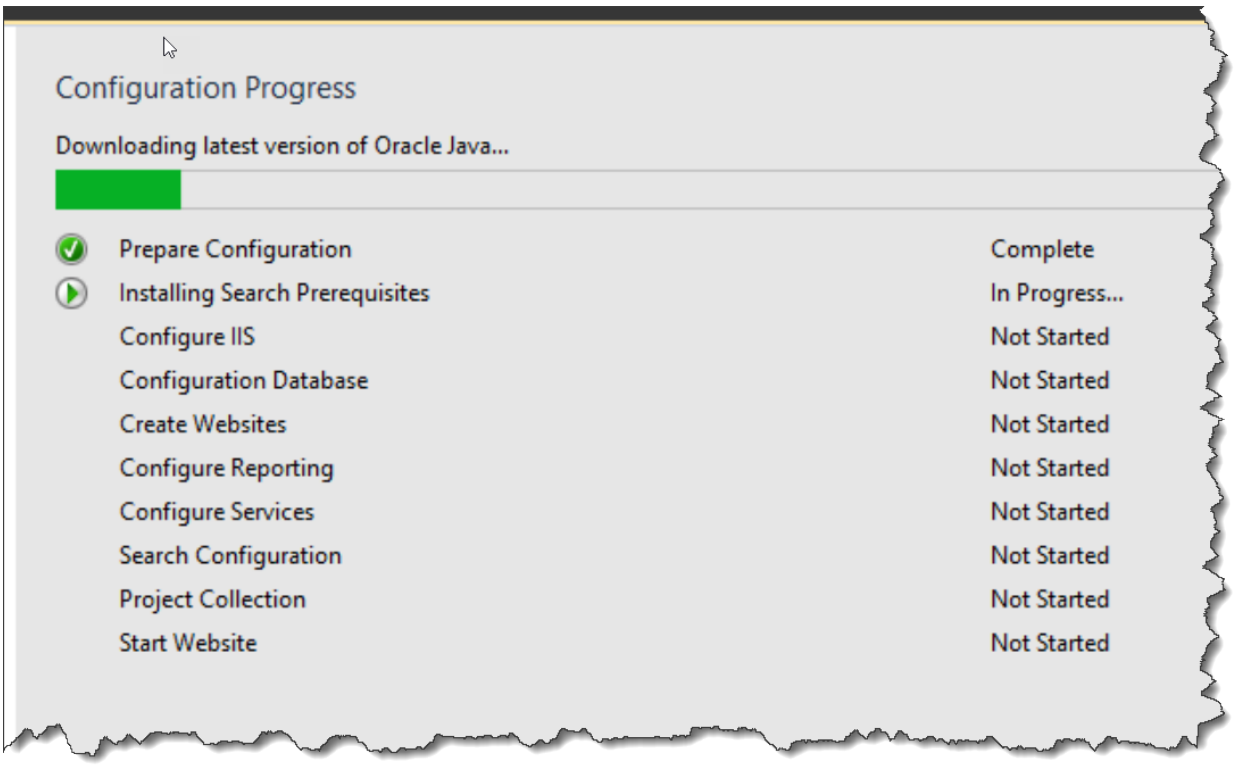
- Click **Next**

The installer will run some readiness checks. They should all come back as passed except for the Search Configuration item. The Search feature requires the Oracle Server JRE to be installed and this warning is prompting you to accept the licensing agreement for the JRE.

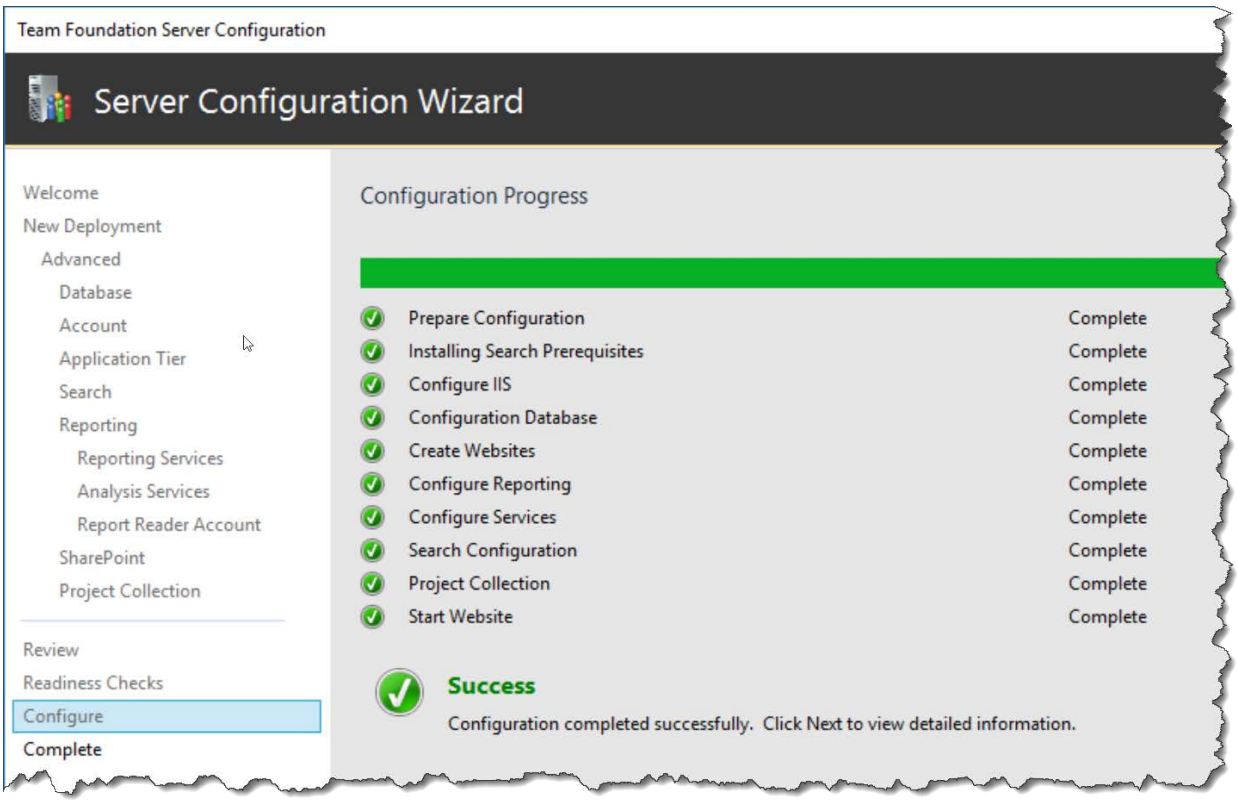


- Check **I accept the Oracle Binary Code License Agreement for Java SE…**
- Click the **Configure** button

The configuration process should now be running.

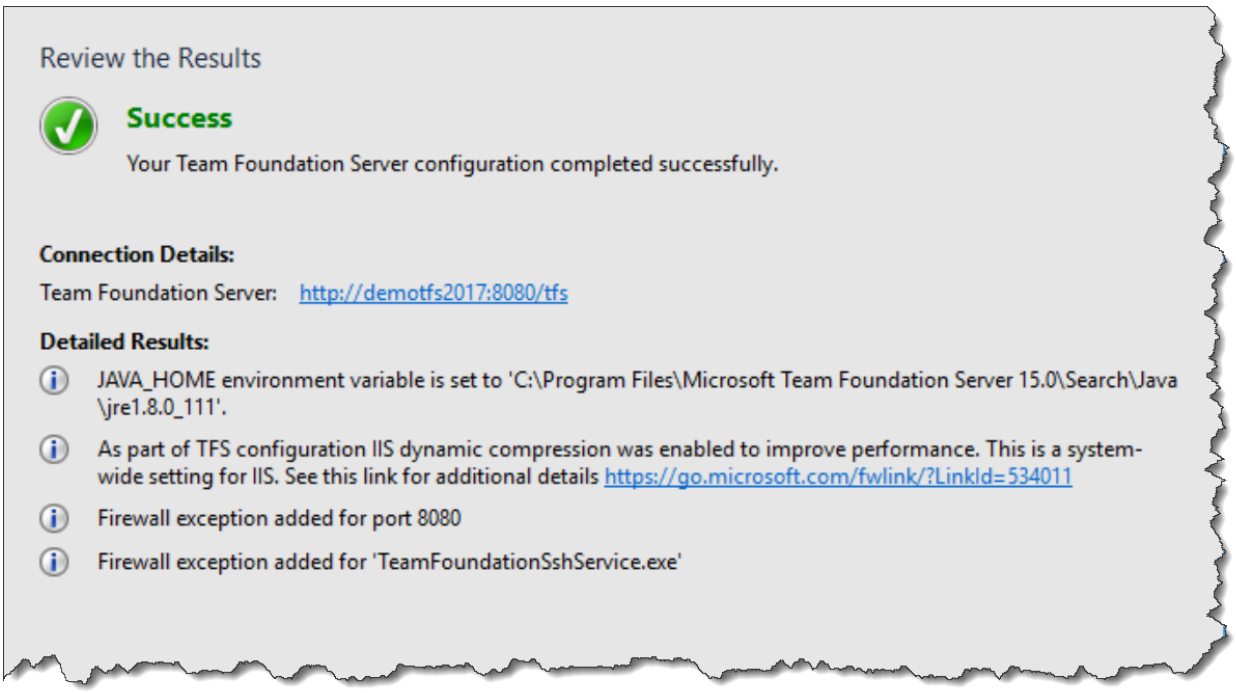


The configuration process should end with a message saying Success.



- Click **Next**

You should now be on the **Review the Results** page.



- Click **Close**

Team Foundation Server 2017 is now configured and running.

Chapter 6: Create a TFS Build / Release Server on Windows

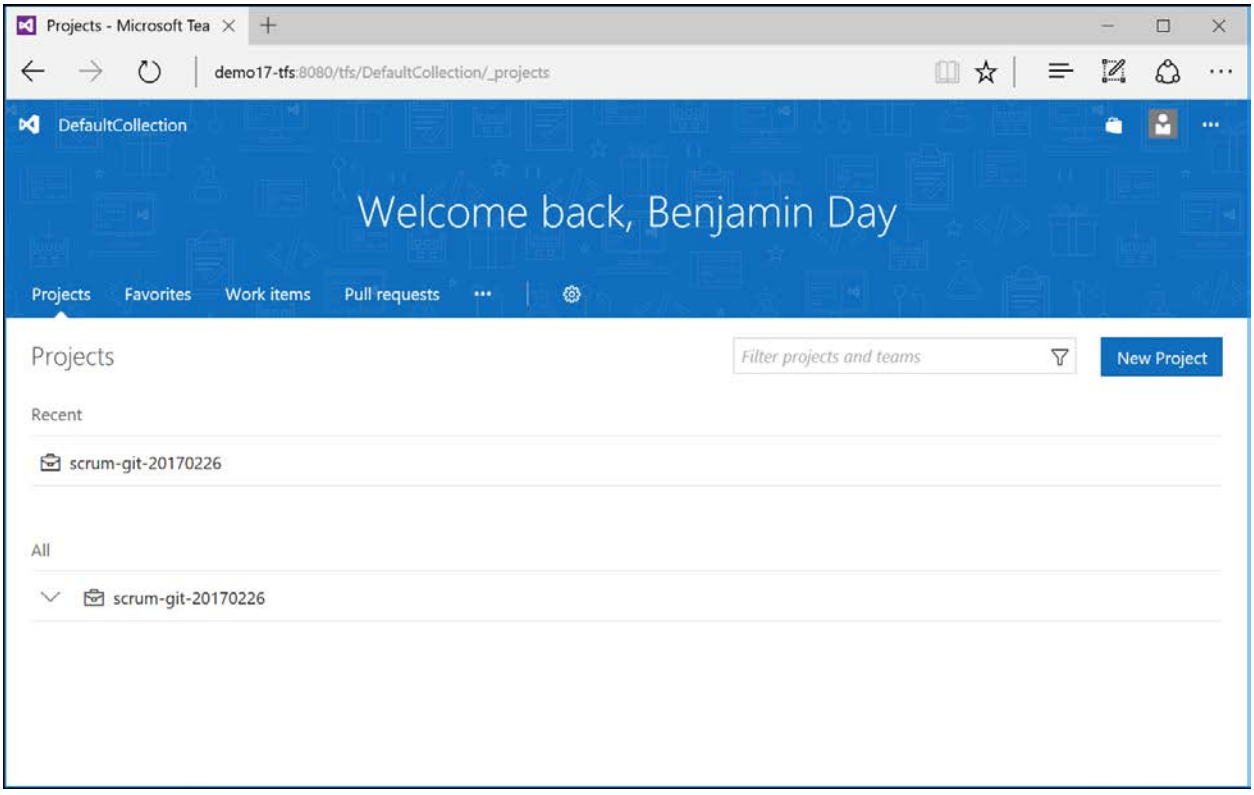
This chapter will walk you through the process of creating a build server on Windows. That's a slightly inaccurate sentence – this chapter will walk you through the process of installing a build & release agent on Windows. The build agent and the release agent are the same installer and process in TFS2017 and a single installation of this agent will allow you to do "build" activities and also "release" activities.

The following steps all happen on the machine that is going to be the build server.

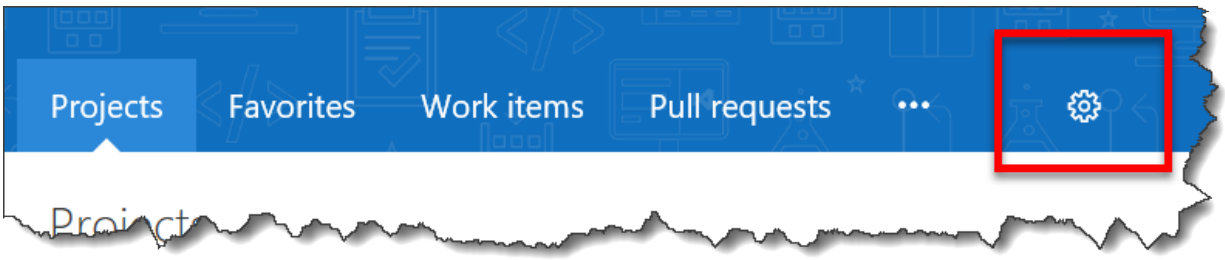
Download the Agent Installer

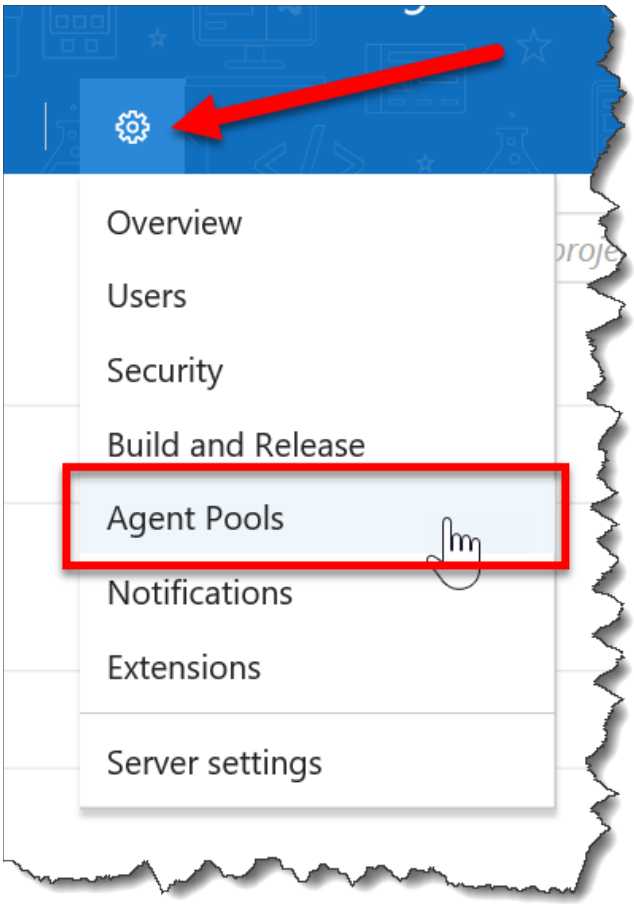
- Log in to the build server machine
- Open a web browser
- Navigate to your TFS web interface. By default this is http://*servername*:8080/tfs.

You should see a screen that looks like this.



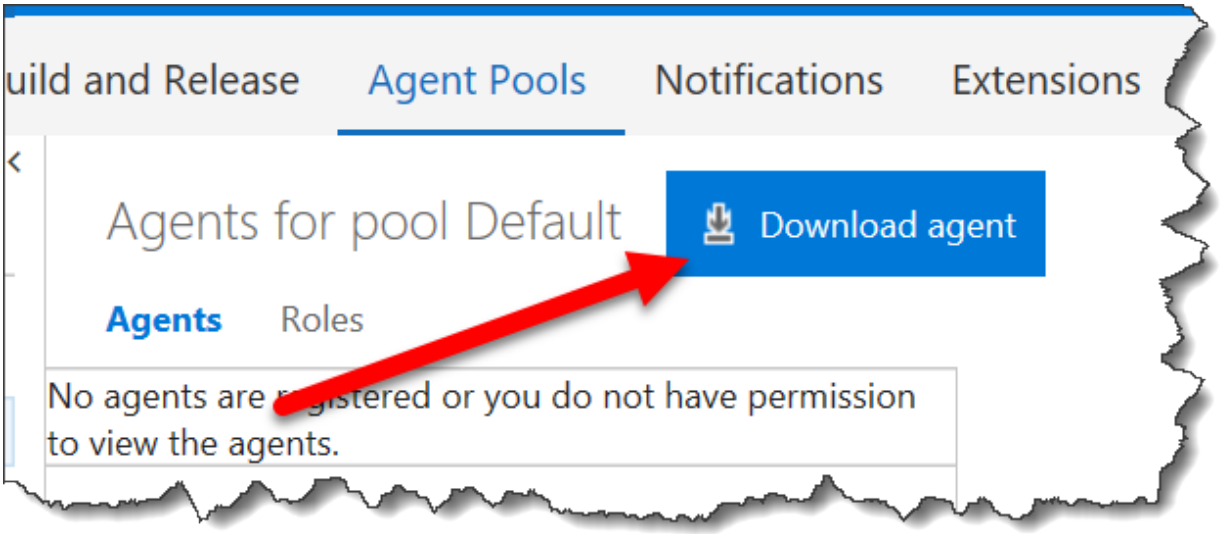
On the right side of the menu bar, there is a gear icon.





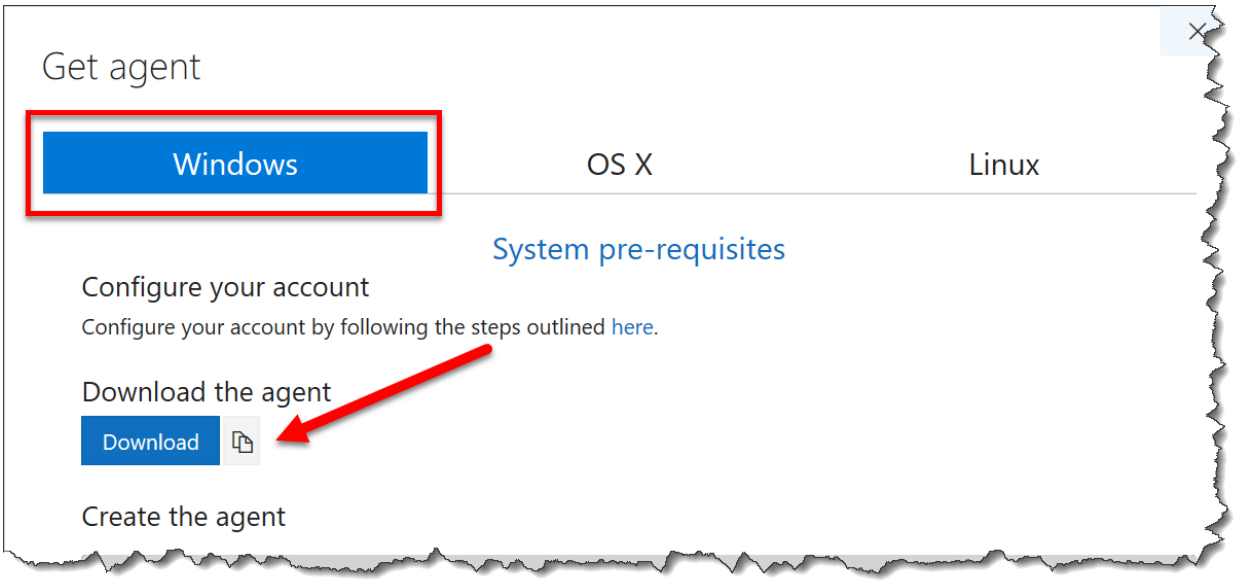
- Click the **gear icon** to bring up the **Settings menu**
- Choose **Agent Pools**

You should now see the admin screen for you team project collection and you should be looking at the Agent Pools tab. There should be a button that says "Download agent".



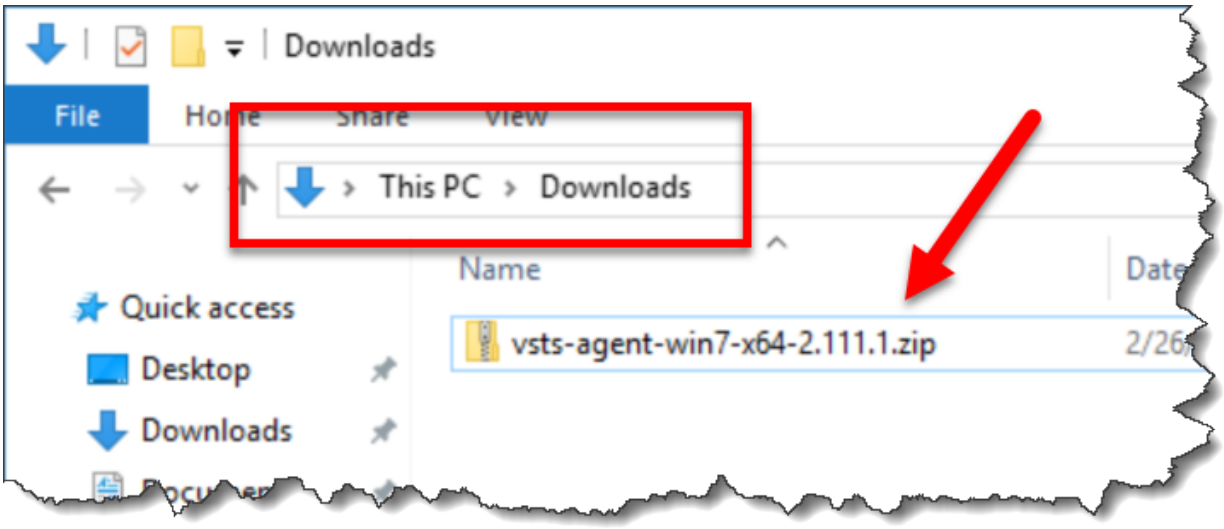
- Click the **Download agent** button

A dialog should pop up that shows you the download and installation info for Windows, Mac OS / OS X, and Linux.



- Make sure the **Windows tab** is selected
- Click the **Download** button (NOTE: this guide assumes you're doing the default browser behavior and downloading the agent zip to the download directory for your user profile.)

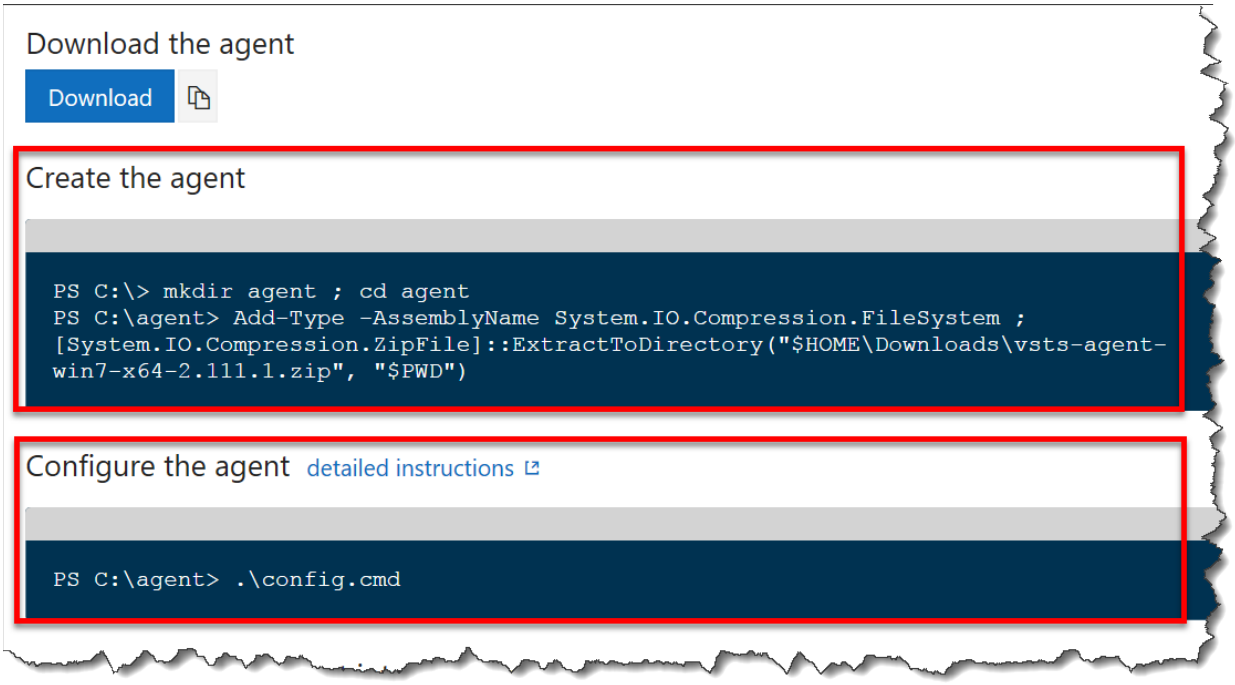
To verify that you saved this file to the expected location for this guide, open Windows Explorer (explorer.exe) and navigate to the Downloads directory. You should see a zip file that has a name that starts with "vsts-agent-". The actual filename might not be the same as the image below because the agent version may have been updated.



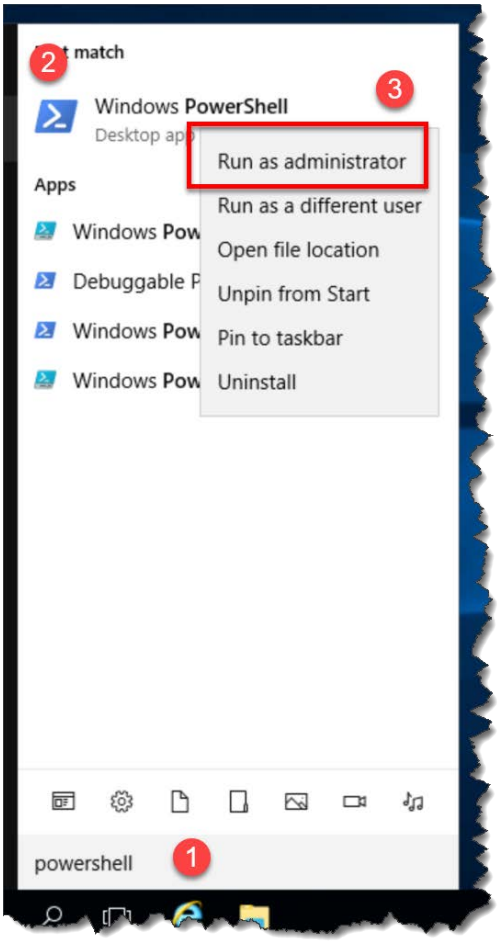
Extract the Agent

Now that you've downloaded the bits for the agent, you'll do the actual installation using PowerShell. This is actually done in two parts. Part 1: Extract the agent bits from the ZIP. Part 2: Configure the Agent. Let's do the first part.

That dialog that you used to download the agent has two different commands: "Create the agent" and "Configure the agent". In an ideal world, you'd be able to just copy and paste the commands and not have to think about anything – but this isn't an ideal world so buckle up. (Actually, it's not that bad…it's just enough friction to be a little bit annoying.)



First up, it's time to run PowerShell.

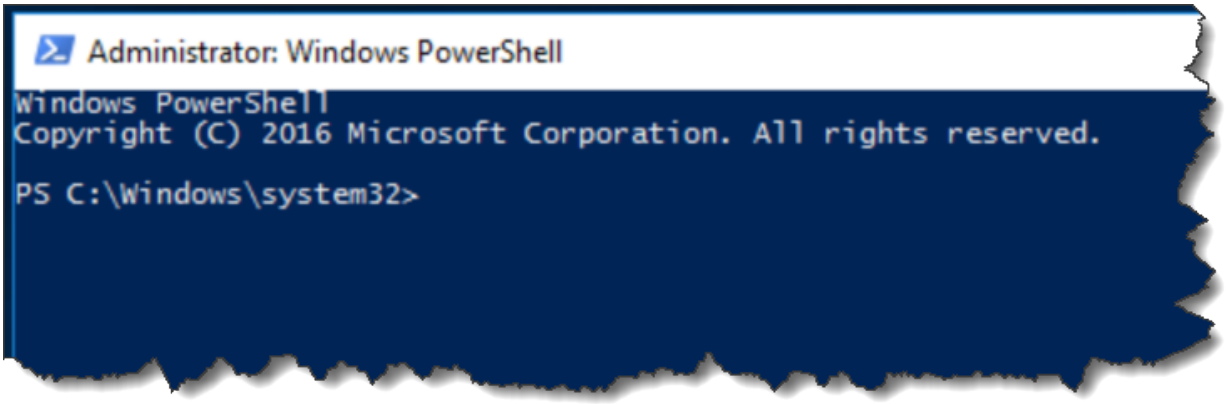


- Press the **Windows key** on your keyboard to bring up the search menu and type **PowerShell**
- From the search results, right-click **Windows PowerShell**
- From the context menu for PowerShell, choose **Run as administrator**

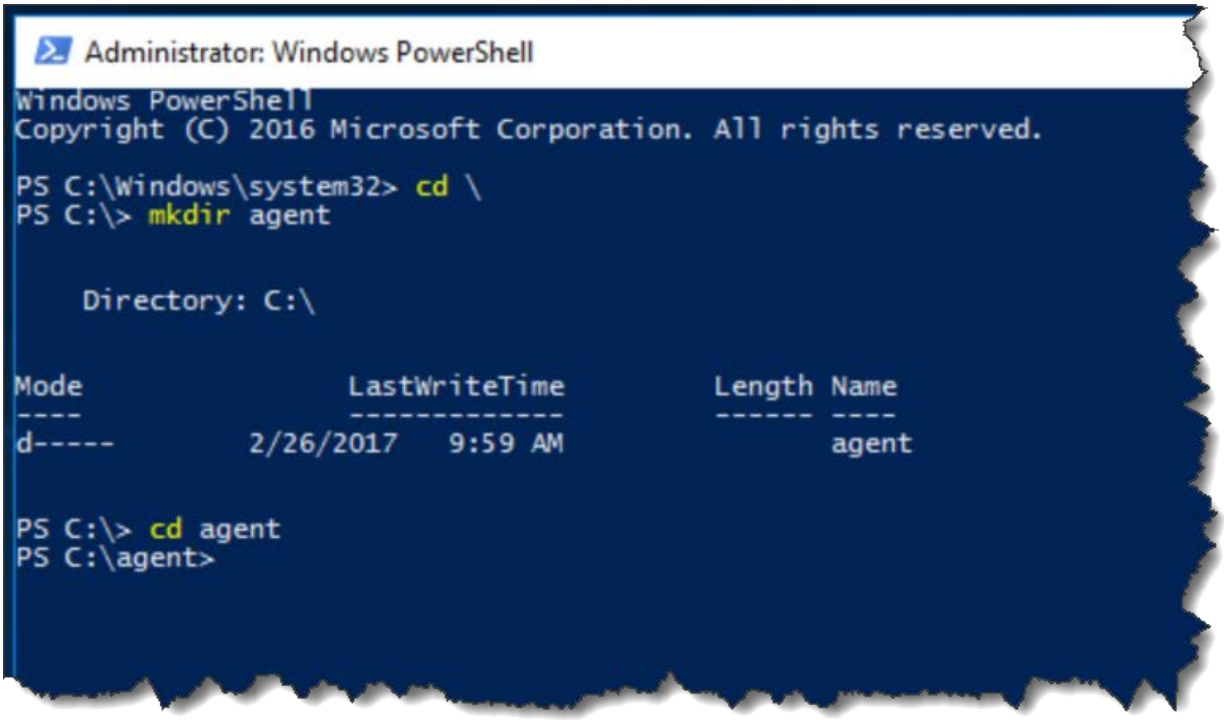
You'll see a User Account Control dialog.

- Click **Yes**

You should now see a Windows PowerShell window with the title "Administrator: Windows PowerShell".



You're now going to run a handful of commands to create the folder structure that you'll be installing the build agent in to.



- Type "cd \" and press Enter
- Type "mkdir agent" and press Enter
- Type "cd agent" and press Enter

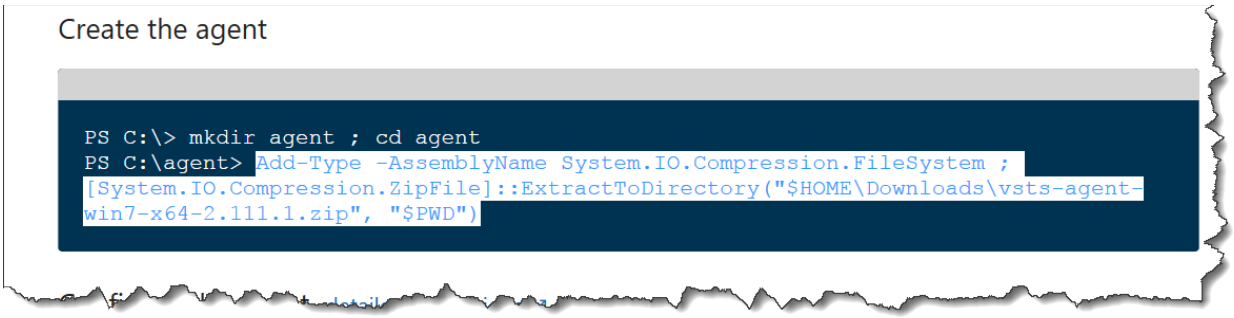
When you're done with these commands, your screen should look almost exactly the same as the image above. It's extremely important that the PowerShell screen is showing you

PS C:\agent>

on the last line because this indicates that you've correctly created a directory called Agent and entered that directory.

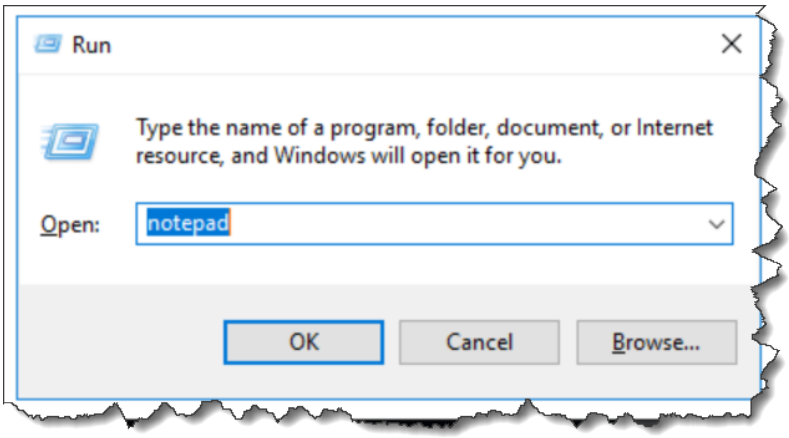
If your PowerShell window doesn't show you "PS C:\agent>", abandon all hope and reconsider all of the life choices that you've made to date. ☺

Next, you'll run the PowerShell command that will extract the zip into the agent directory. This is some fiddly typing and the exact text will change as Microsoft updates the build agent install zip filename. It's probably easiest to just copy and paste the value from the **Create the agent** section of the download dialog. You WILL NOT be copying the whole command. You'll only be copying part of the line. In the image below, it's important to notice that I am NOT selecting the portion of the line that starts with "PS C:\agent>".



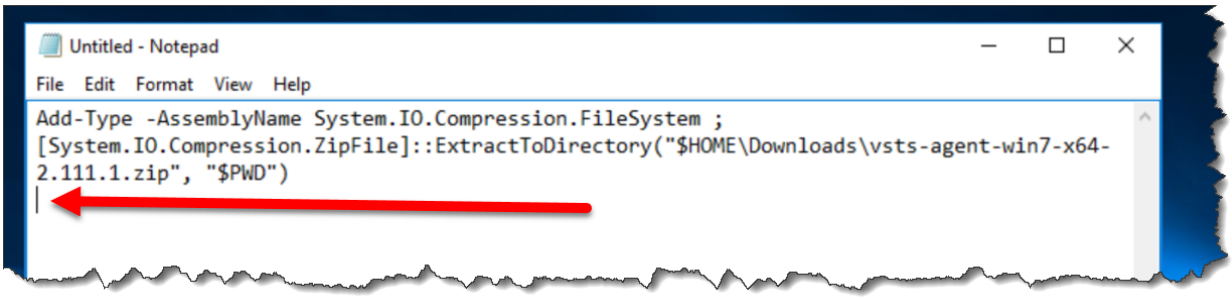
- In the web browser, copy the text of the command that starts with "Add-Type"

Because nothing is ever easy, the command that you just copied probably has some extra characters in it. Let's use Notepad to fix it.



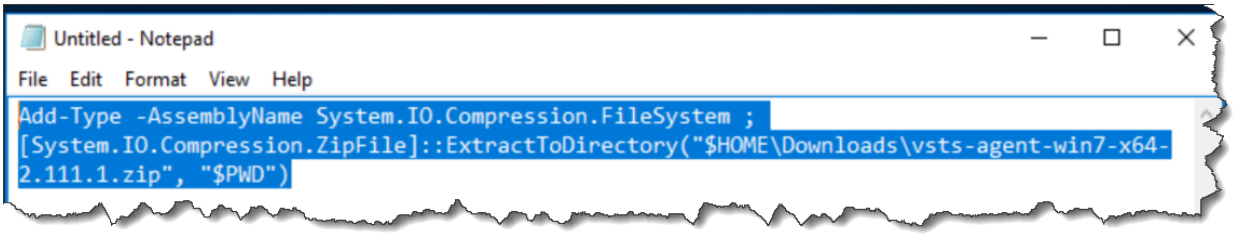
- Type Windows-R to bring up the Run dialog
- Type notepad
- Click OK
- Paste the copied command into Notepad

You should now see the command in Notepad. If everything is on one long line, go to the Format menu and choose Word Wrap. You might notice in the image below that my cursor is sitting on an empty line by itself. It's this extra line that's causing us to do this Notepad step.



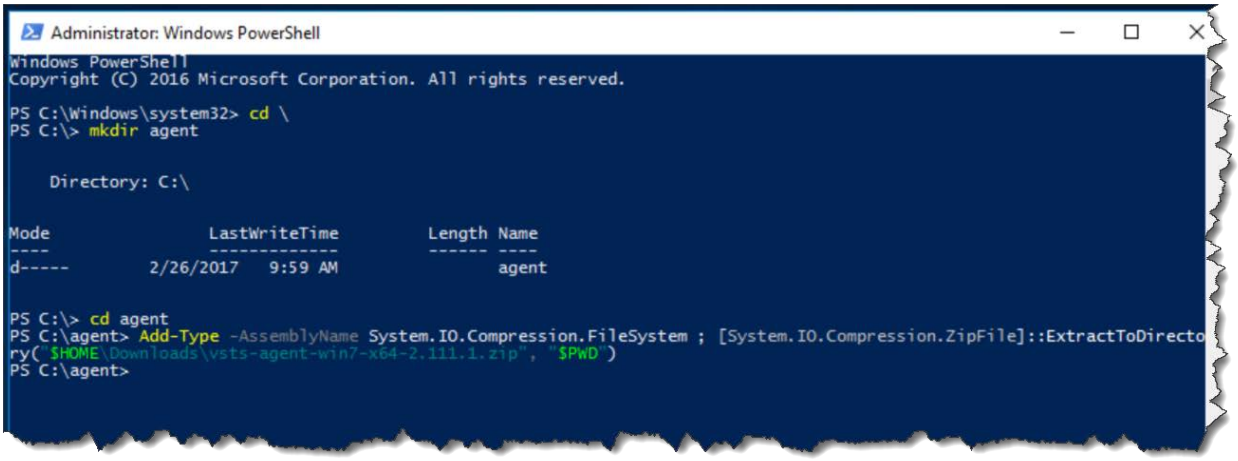
- Delete the extra empty blank line at the end
- If there are any whitespace characters before "Add-Type", delete those, too.

Now that that extra characters are gone, you should have a clean command that you can just paste into PowerShell and run.



- Select the command
- Press CTRL-C to copy the selected command to the clipboard

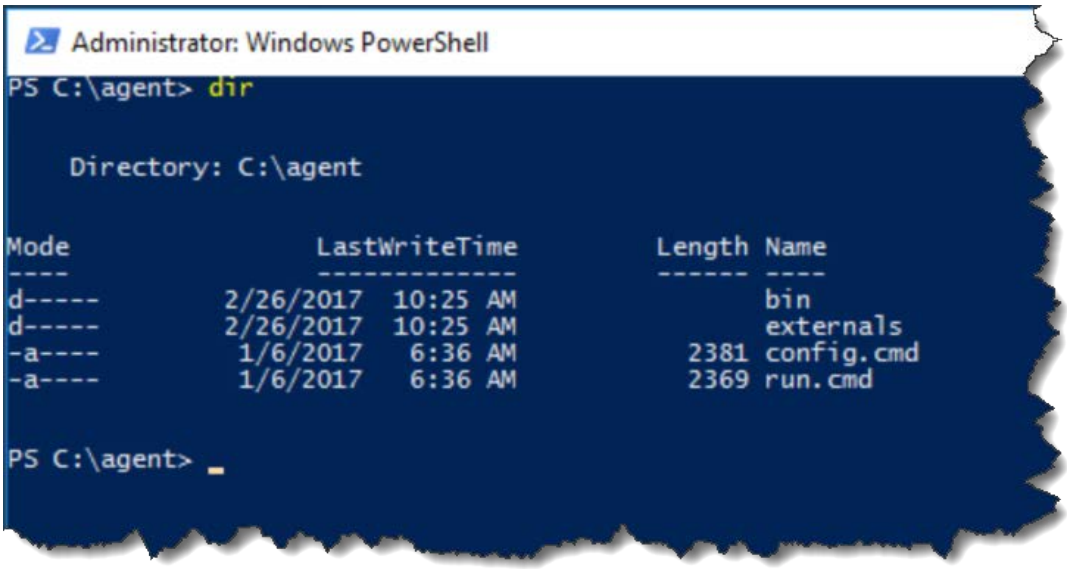
Now you'll run the command in PowerShell.



- In the PowerShell window, type **CTRL-V** to paste the command
- Press **Enter** to run the command

When the command is done, you should not see any errors and the prompt should say "PS C:\agent>".

Let's verify that this extracted as expected.



- (Optional) To clear the screen, type "cls" and press Enter
- Type "dir" and press Enter

The screen should look something like the image above.

Configure the Agent

Now that the agent bits are deployed to disk, you're ready to start configuring it. This guide assumes that you're planning to run this agent in a Windows domain and that the TFS machine is in the same domain as the agent. I'm also assuming that you intend to run this agent as a service rather than as an interactive process.

Recommendation: The agent can be configured to run as NT AUTHORITY\NETWORK SERVICE but I think that this makes permissions management confusing when you're creating and running builds. I strongly recommend that you run the agent as a service using a known service account that is based on an Active Directory user rather than one of the build-in service accounts like NETWORK SERVICE. This guide will assume that you're following this recommendation.

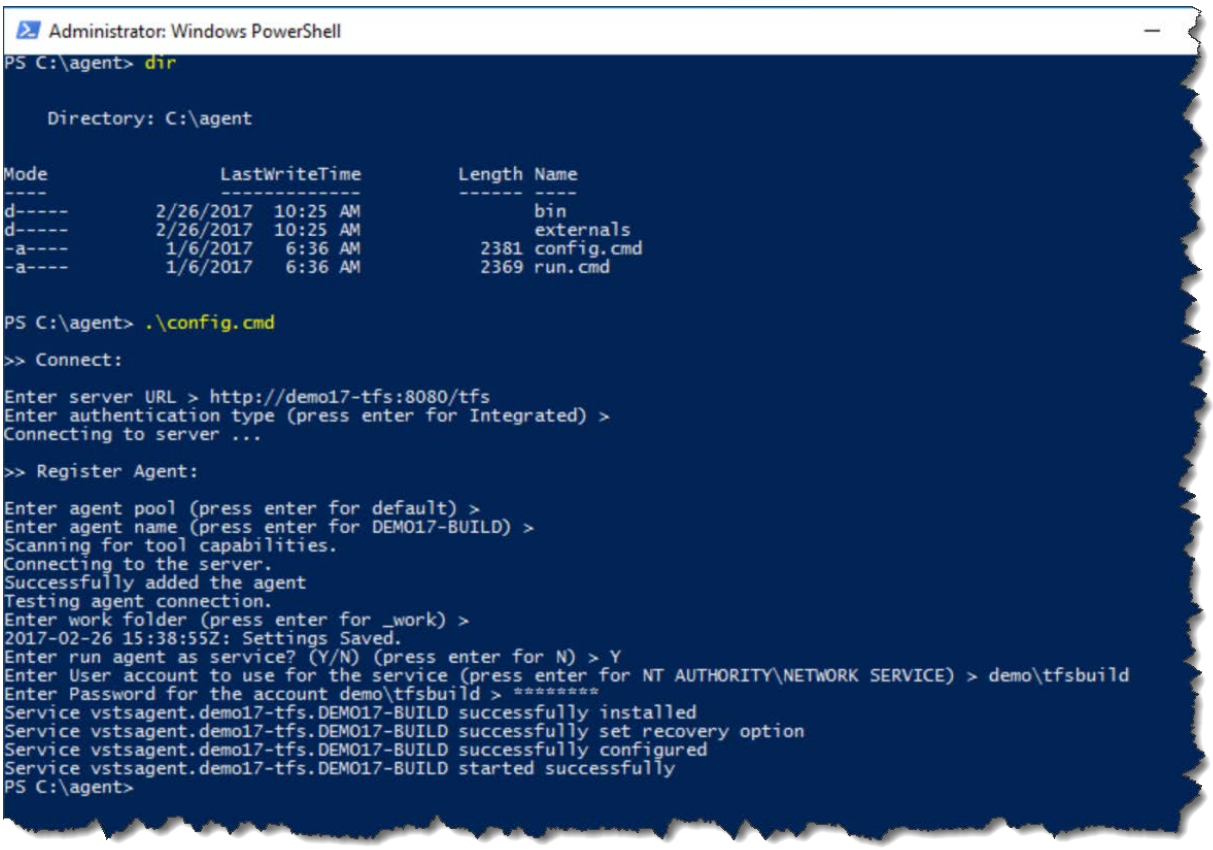
You're going to be prompted for a handful of values during the configuration process:

- **TFS server URL:** This is the same URL that you used to access the TFS web interface. By default this will be something like http://*servername*:8080/tfs
- **Authentication Type:** By default, authentication is based on the service account's Windows logon. This mode is called Interactive. In order support more complex scenarios and multiple platforms, there are also several other options. This guide will show you how to do Interactive mode.
- **User name & password for the agent service:** These are the credentials for the service. In my case, I've created an Active Directory user named "tfsbuild". The fully qualified username for this user is "DEMO\tfsbuild".

When you've got these values, you're ready to run the config process.

- In the PowerShell window, type "**.\config.cmd**" and press Enter

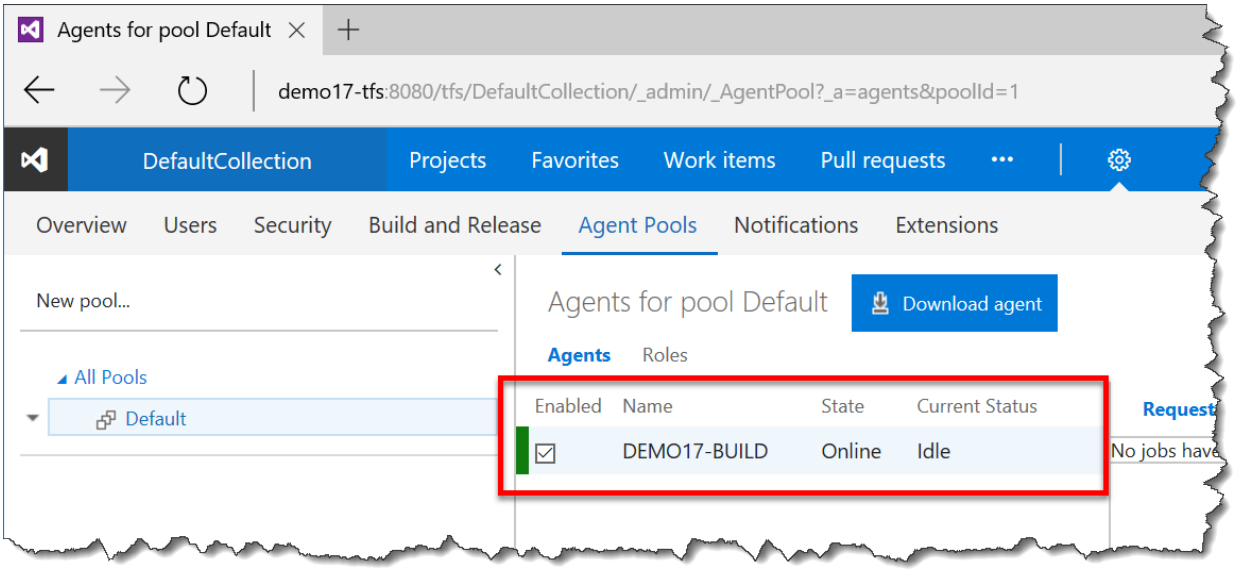
When prompted, enter the following values.



- "Enter server URL":
  Type the **URL for your TFS instance** and click **Enter**
- "Enter authentication type (press enter for Integrated)":
  Press **Enter**
- "Enter agent pool (press enter for default)":
  Press **Enter**
- "Enter agent name (press enter for *[local server name]*)":
  Press **Enter**
- "Enter run agent as service? (Y/N)":
  Type '**Y**' and press **Enter**
- "Enter User account to use for the service":
  Type the **fully qualified name of the service account** (example: demo\tfsbuild) and press **Enter**
- Enter Password for the account *[service account]*":
  Enter the **password for the service account** and press **Enter**

When the config process has completed, you should see a message that says something like "Service vstsagent.demo17-tfs.DEMO17-BUILD started successfully".

If you open the browser and go back to the Agent Pools tab for TFS, you should now see your new build agent in the list of Agents.



You've successfully configured a build agent.