How To: Install SQL Server 2016 – Failover Cluster Instances

Version 0.2

# Version History

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Author | Reason |
| 0.1 | 28/12/2017 | Shaun O’Dell | First draft. |
| 0.2 | 01/06/2018 | Shaun O’Dell | Add port configuration |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

This document describes the installation of an instance of SQL Server 2016 Failover Cluster Instance.

For SQL Server Availability Groups see the document “How To: Install SQL Server 2016 – Availability Group”, for standalone installations see “How To: Install SQL Server 2016 – Single Server”.

This document covers only the core SQL Server database engine, SQL Server Agent, and SQL Server Integration Services installations, additional features such as SQL Server Reporting or Analysis servers are not considered.

Before you begin you will need to know several items of information to allow the instance to be installed and configured correctly, these should be specified and agreed on a SQL Server Installation Request. *It is not possible to complete the installation of SQL Server 2016 without these details.*

Full details of the SQL Server Installation Request form, explanatory notes for its completion, recommendations and best practice can be found in “*How To: Complete a SQL Server Installation**Request”*.

Note that beginning with the release of SQL Server 2016 the Management Tools feature is no longer available as a feature within the SQL Server setup; SQL Server Management Studio is available as a separate installation which must be used if required.

# Prerequisites

* Completed and agreed SQL Server Installation Request
* Cluster built with Windows Server and patched
* *A valid SQL Server 2016 licence*
* Access to appropriate SQL Server 2016 installation media (a copy can be found at \\SQLDBA\Media$\SQL Server 2016)
* Access to SQL Server 2016 Service Packs and Cumulative Updates (copies are available at \\SQLDBA\Media$\SQL Server 2016)
* Service accounts created and replicated across the domain
* An IP addresse for each SQL Server instance to be installed
* Optionally, an IP address foe the per-instance MSTDC

## Disk Layout Considerations

When building an instance of SQL Server, it should have four disks provisioned for SQL Server database files, if the target server is a virtual machine these should be provisioned as separate disks to allow for future resizing rather than as partitions. If the target server is physical it acceptable to create the SQL System disk as a partition of the user database disk. In addition, it is expected that separate disks will be provided for the OS, Applications, and, possibly the operating system page (swap) file.

For a single-instance failover cluster the drive letters used should follow the standard set out below, for a multiple-instance failover cluster different drive letters will be required for .

|  |  |  |
| --- | --- | --- |
| Drive Letter | Use |  |
| C | Operating System | Locally attached storage |
| D | Swap file | Locally attached storage |
| E | Applications | Locally attached storage |
| F | User database data **f**iles | SAN shared storage |
| L | User database transaction **l**ogs | SAN shared storage |
| S | System databases and logs | SAN shared storage |
| T | TempDB | Locally attached *or* shared SAN storage, see below. |

The operating system and application drive, and swap disk if present, must use locally-attached storage.

The disks used for SQL Server system databases, user database files, and user database transaction log files must use share, SAN-attached storage.

Starting with the release of SQL Server 2012 it is possible to host TempDB either on shared storage (as was the case in previous versions of SQL Server) or on fast, locally-attached storage – for example using SSD drives.

In the case of locally-attached storage the same drive letter(s) and paths **must** be present on each node of the failover cluster which is to host the SQL Server instance

## Service Account Considerations

Failover cluster installations require the use of domain accounts, virtual accounts are not supported. Please note that Managed Service Accounts (MSA) and Group Managed Service Accounts (gMSA) are not currently supported at Yorkshire Water.

The active directory accounts used for the SQL Server and SQL Server Agent services, should follow the following naming convention and be created with strong, complex passwords. The “**Password never expires**” and “**User cannot change password**” boxes must be checked when creating service accounts.

* <Domain>\SQLSvc\_<Instance> for the SQL Server Database Engine service
* <Domain>\SQLExec\_<Instance> for SQL Server Agent

<Instance> is the instance name for a named instance of SQL Server or the host computer name for a default instance of SQL Server.

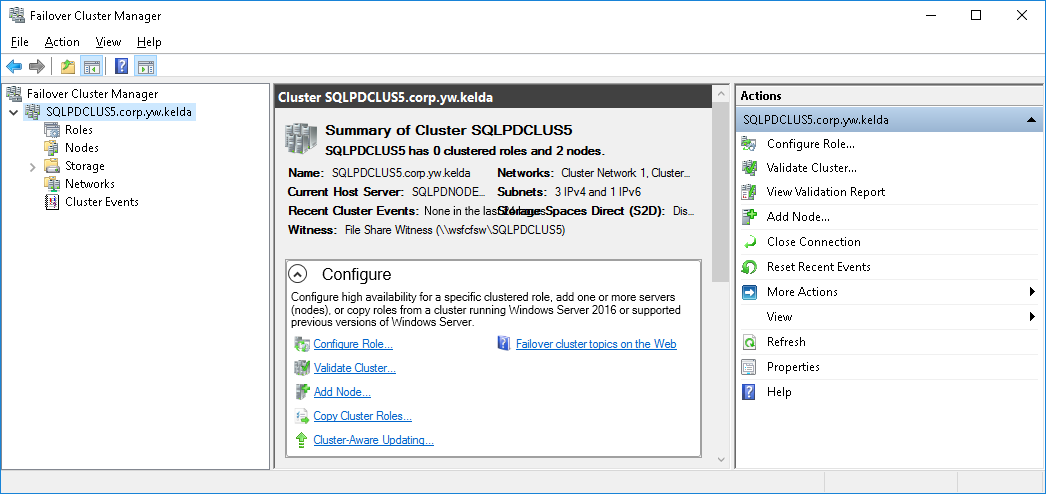
Do not add the service accounts to the local Administrators group of the target servers for the failover cluster installation.

For further details see <https://docs.microsoft.com/en-us/sql/database-engine/configure-windows/configure-windows-service-accounts-and-permissions>

## Cluster Validation

It is important that the cluster is validated before SQL Server is installed. To validate the cluster:

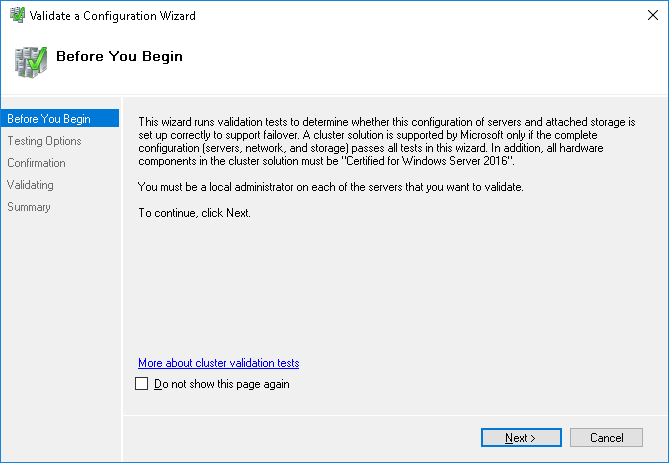
Start Failover Cluster Manager



Select the core cluster resource from the Console Tree pane on the left-hand of the console

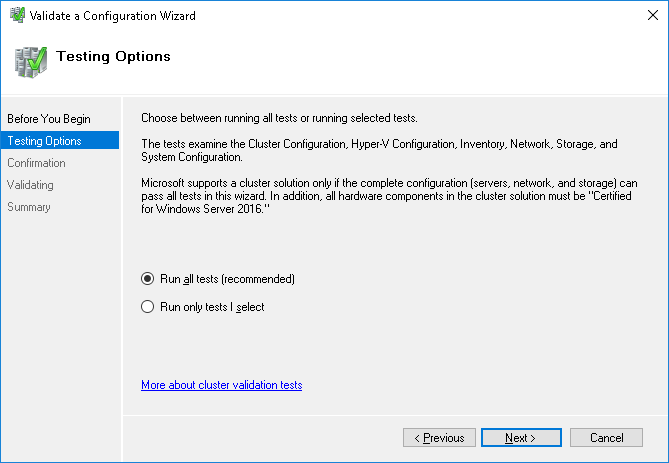
Either click **Validate Cluster…** from the right-hand **Actions** pane or right-click the cluster resource and select **Validate Cluster…** from the context menu.

The **Before You Begin** page is displayed



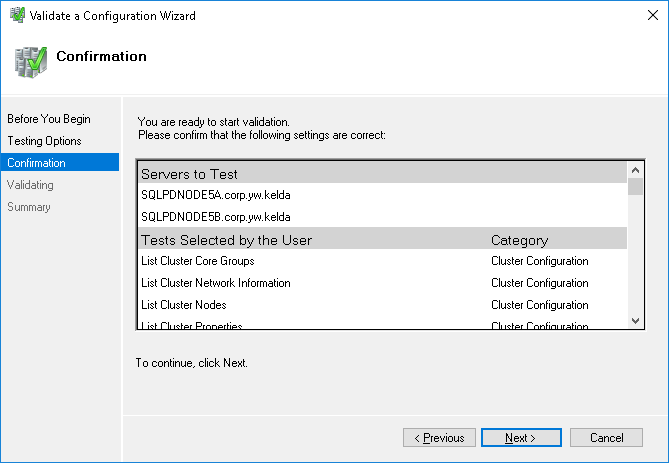
Click **Next**

The **Testing Options** page is displayed.

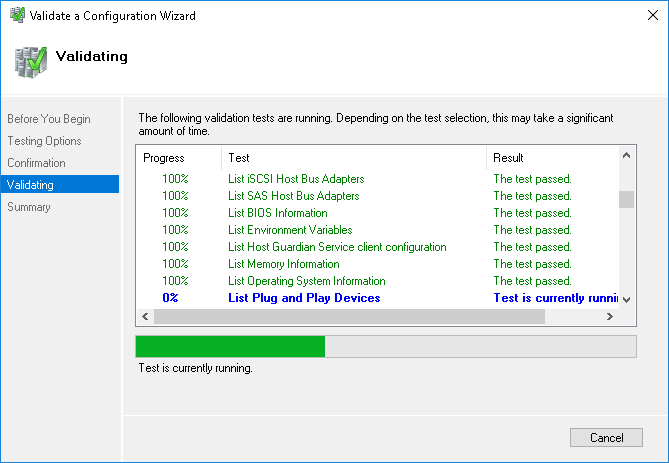


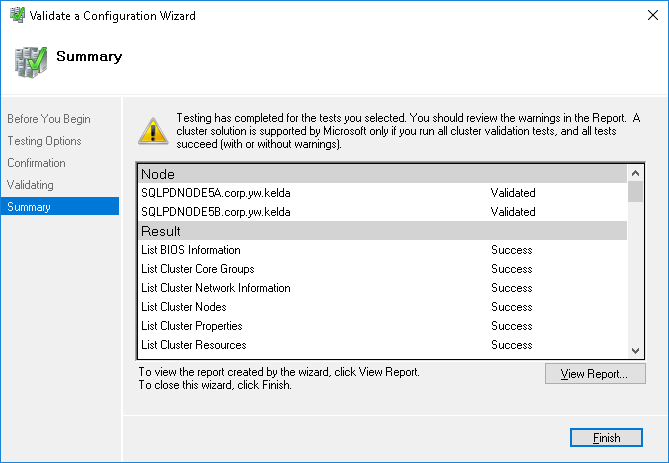
Select Run all tests (recommended) and click Next

The **Confirmation** page is displayed



Click **Next**





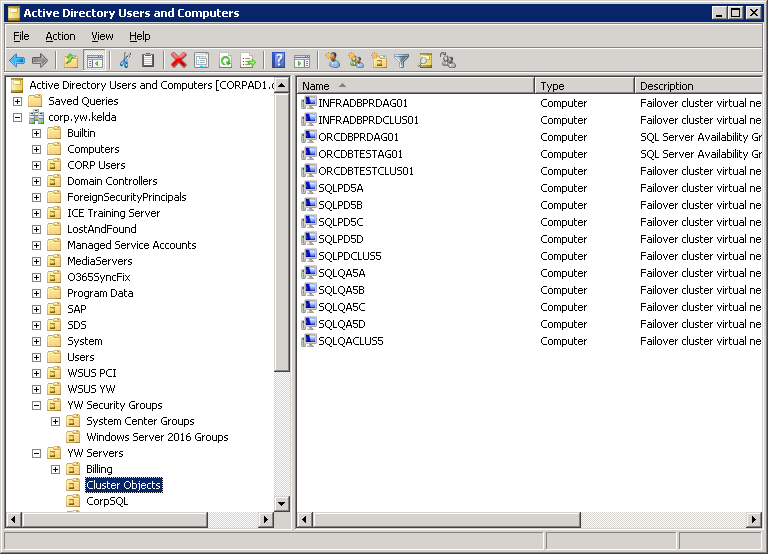
## Configure the CNO in Active Directory

Before proceeding it is essential for Windows Server 2016 based cluster that the Cluster Name Object or *CNO* (for example SQLPDCLUS5) is moved to the **Cluster Objects** organizational unit (corp.yw.kelda/YW Servers/Cluster Objects). This ensures it has the rights to create and manage the Cluster Virtual Objects within AD and DNS.

Log on to a domain controller for corp.yw.kelda

Locate the Cluster Name Object, for example SQLPDCLUS5

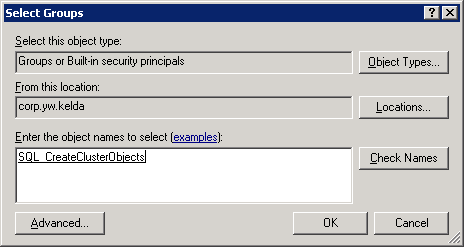
Move the CNO to the *Organizational Unit* **Cluster Objects** under **YW Servers**



Once moved locate the CNO in the **Cluster Objects** OU.

Right-mouse click the CNO and from the context menu select **Add to a group…**

Enter the group name **SQL\_CreateClusterObjects** in the Select Groups dialog and click **Check Names**



Click **OK**

For reference the *SQL\_CreateClusterObjects* group can be found at YW Security Groups/Windows Server 2016 Groups/SQL\_CreateClusterObjects

# Install SQL Server 2016

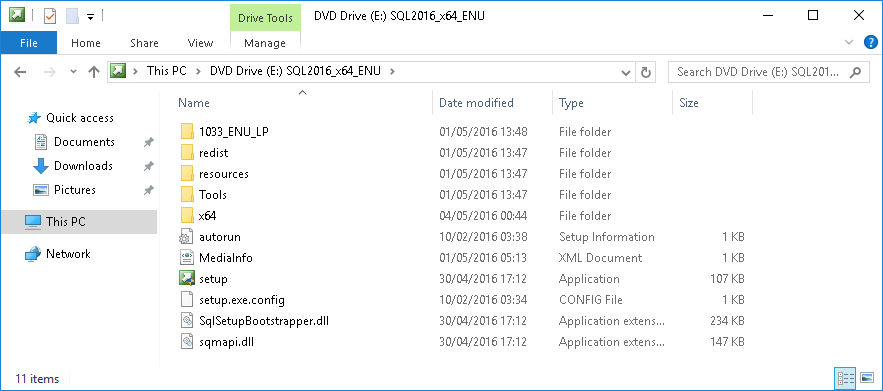
## First Cluster Node

Log on to the first node of the cluster with Administrative rights.

### Mount the SQL Server installation ISO image

On the first cluster node, mount the appropriate SQL Server 2016 ISO:

* For MSDN licensed installations media are located at \\SQLDBA\Media$\SQL Server 2016\MSDN
* For Production licensed installations media are located at \\SQLDBA\Media$\SQL Server 2016\Volume Licence



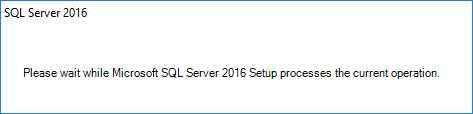
### Setup SQL Server 2016

Launch Setup.exe using the right-mouse context menu and select “**Run as administrator**”

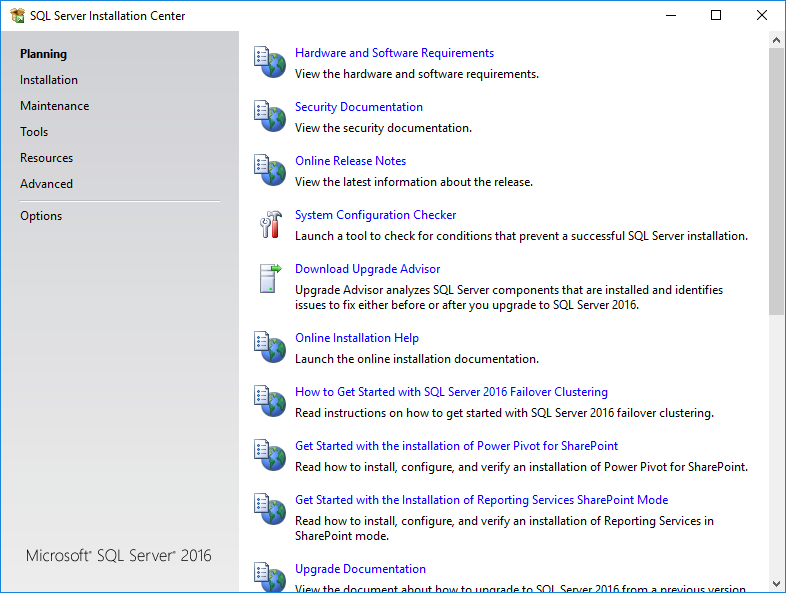
If prompted by User Account Control “Do you want allow this app to make changes to the device?” (as illustrated below), click **Yes**



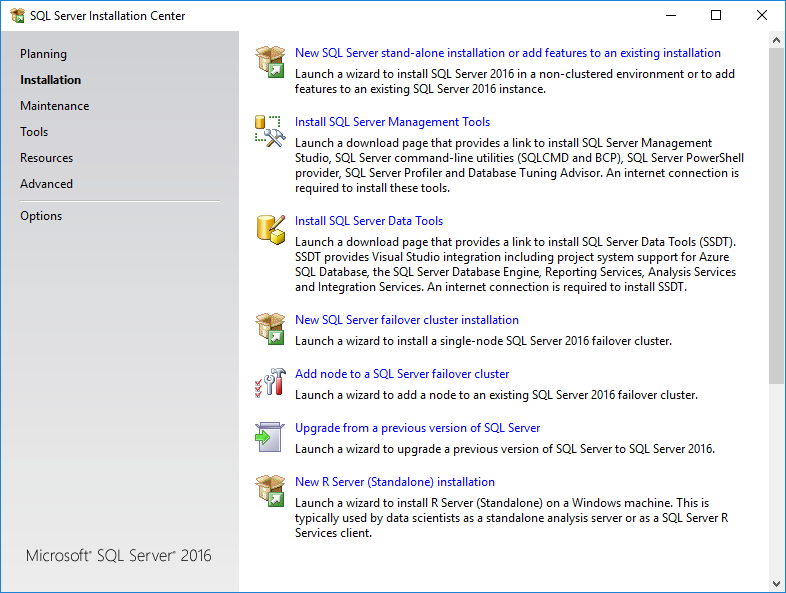
The setup will launch and display the following message:



After a few seconds the **SQL Server Installation Center** will be displayed:

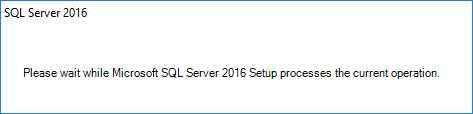


Select **Installation** from the left-hand side of the **SQL Server Installation Center**



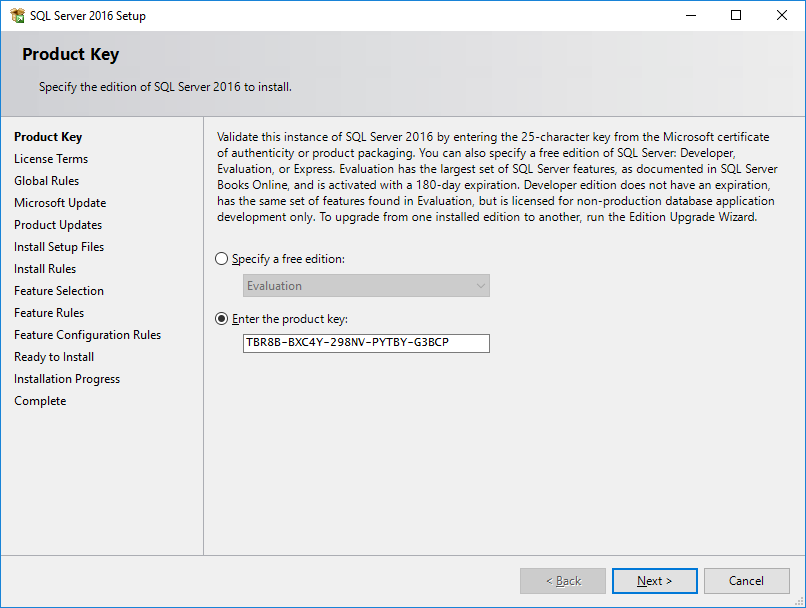
Click **New SQL Server failover cluster installation**

The following message will be displayed for a few seconds whilst the setup is loading



From this point on the order of pages shown is based on Microsoft’s documentation and applies to the *first* install of SQL Server. In a multi-instance installation experience shows that the installation wizard will bypass the first two pages, **Product Key** and **Licence Terms** and start at **Global Rules**. In this case **Product Key** and **Licence Terms** will be displayed after the **Install Failover Cluster Rules** page and before the **Feature Selection** page.

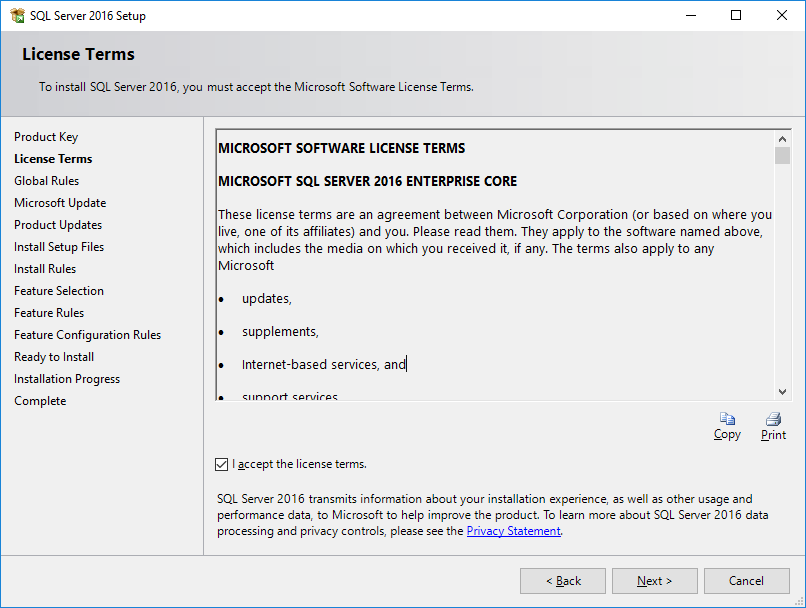
The **SQL Server 2016 Setup** wizard will be displayed at the **Product Key** page



The product key will already be present if using media from the SQL Server Media share (\\SQLDBA\Media$)

Click **Next**

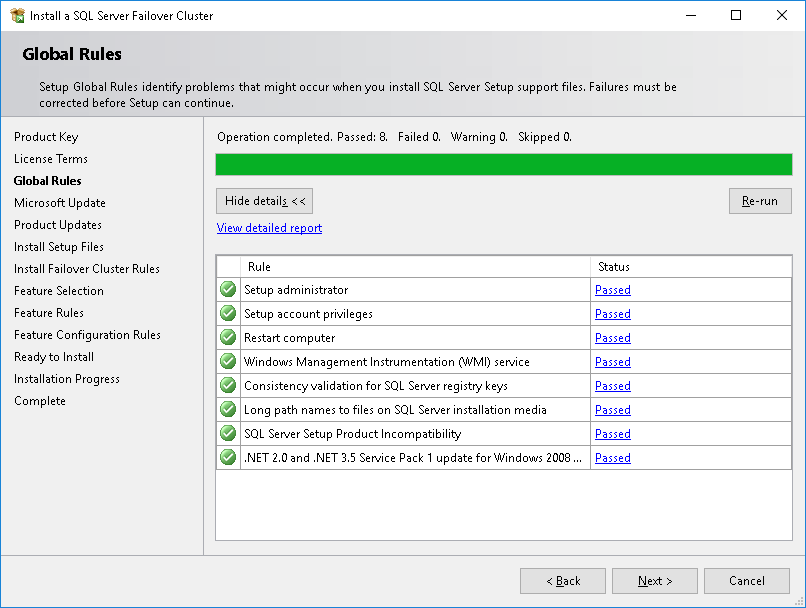
The **License Terms** page is displayed



Check “**I accept the license terms**” and click **Next**

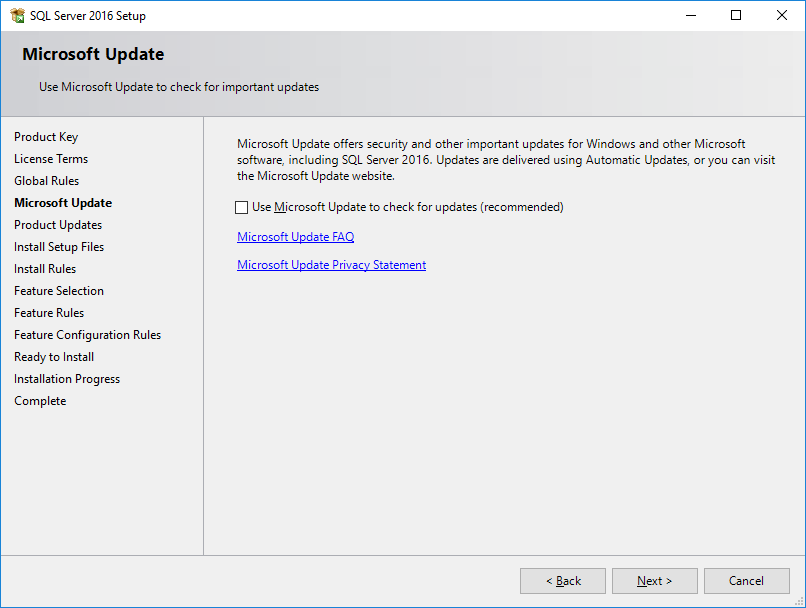
The **Global Rules** page is displayed.

Whilst the Setup Global Rules are being evaluated further details of the progress being made can be viewed by clicking the **Show details >>** button, the screen capture below shows the expanded details.



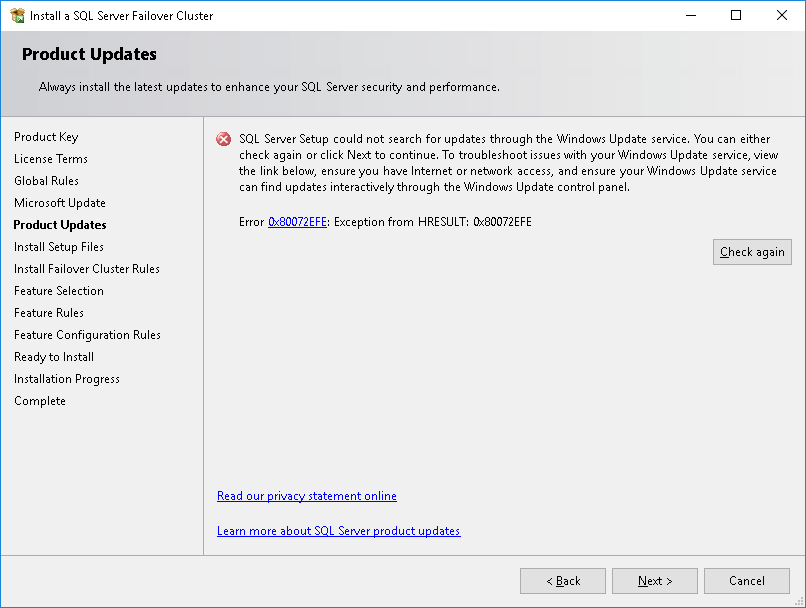
Click **Next** if all rules have passed

The **Microsoft Update** page is displayed



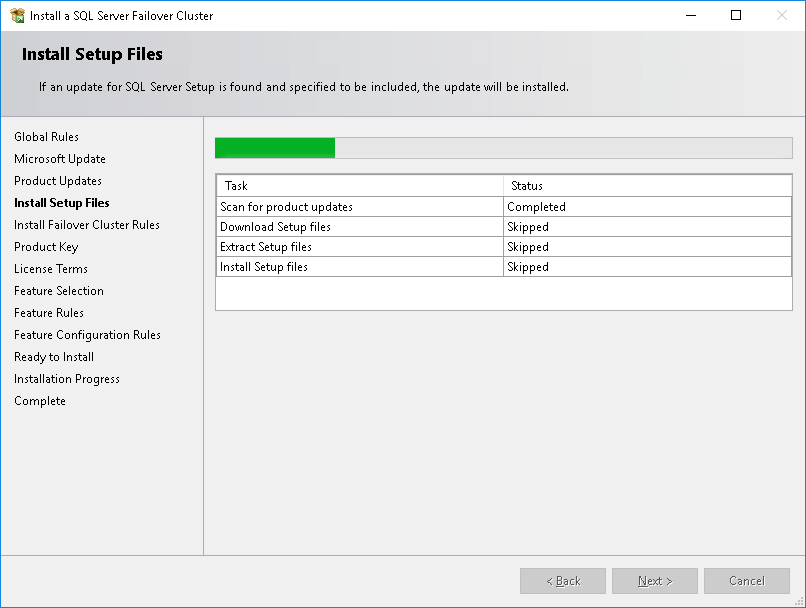
Leave **Use Microsoft Update to check for updated (recommended)** unchecked, updates will be manually applied to allow control of the version of SQL Server built. Click **Next**

Some versions of the setup program will skip the **Product Updates** page whilst some will display **Product Updates** page with the following error.



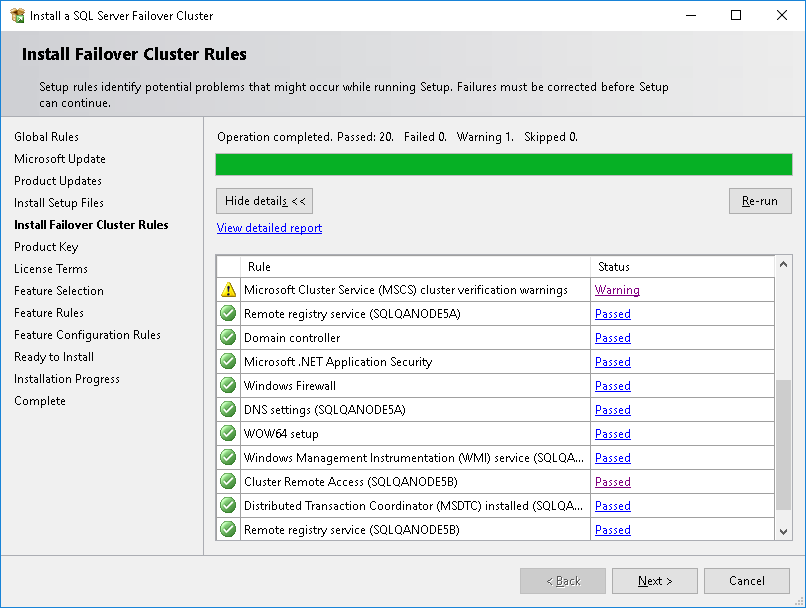
If the **Product Updates** page is displayed with error 0x80072EFE this simply indicates that setup cannot reach the internet to check for updates. As updates are applied manually this can safely be ignored, click **Next**

The **Install Setup Files** page will be displayed



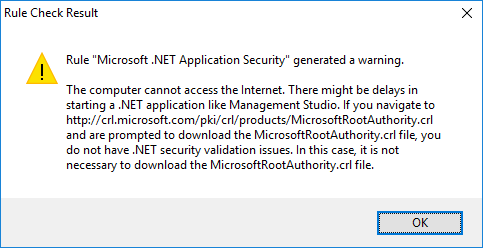
No user intervention is required.

The **Install Failover Cluster Rules page** is displayed



Whilst the Install Rules are being evaluated click **Show details >>** to expand the details of the rules being evaluated. It is highly likely that a Warning will be shown against Microsoft .NET Application Security; this is most likely due to the server not having internet access.

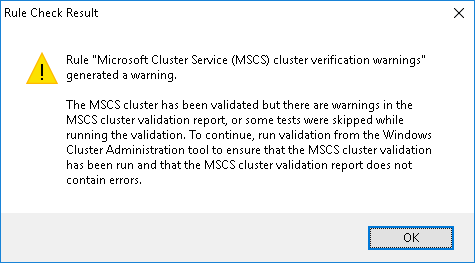
Click the **Warning** hyperlink; the following message is displayed:



Click **OK** to dismiss this. You can carry out a further check by clicking the **View detailed report** hyperlink, if the warning is shown for SSMS\_IsInternetConnected with the following message you may proceed with the installation:

“The computer cannot access the Internet. There might be delays in starting a .NET application like Management Studio. If you navigate to http://crl.microsoft.com/pki/crl/products/MicrosoftRootAuthority.crl and are prompted to download the MicrosoftRootAuthority.crl file, you do not have .NET security validation issues. In this case, it is not necessary to download the MicrosoftRootAuthority.crl file.”

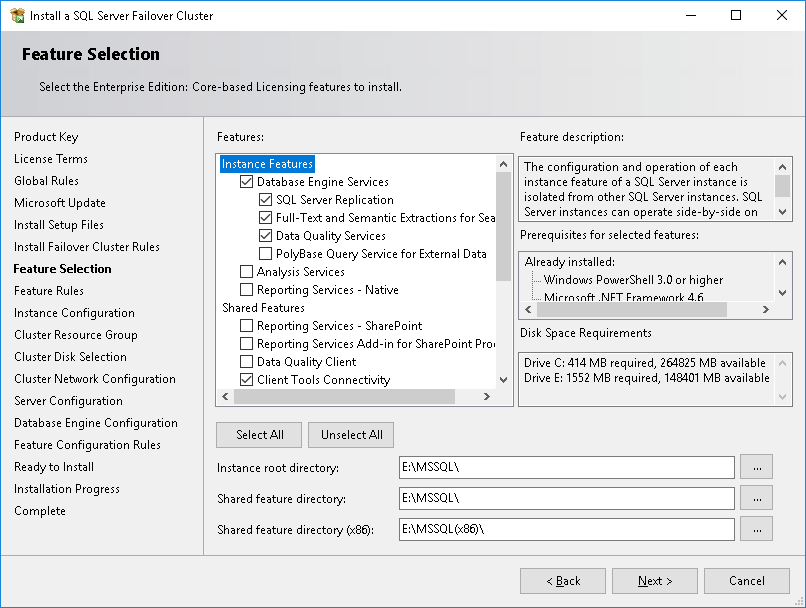
You may also see a Warning for Microsoft Cluster Service (MSCS) cluster verification warnings



Refer to the earlier section on *Cluster Validation,* if all tests were passed or skipped proceed with setup. Any failures or warnings must be investigated and resolved.

Click **Next**

The **Feature Selection** page is displayed



As a minimum select:

* Instance Features
  + Database Engine Services (*for a failover cluster installation SQL Server Replication, Full-Text and Semantic Extractions for Search, and Data Quality Services will be selected if Database Engine Services is chosen, these cannot be de-selected*)
* Shared Features
  + Client Tools Connectivity
  + Client Tool Backwards Compatibility
  + Client Tools SDK
  + Integration Services (if required)

Note that if multiple instances are being installed in the failover cluster that shared features selected for the first instance installed will be automatically selected, and cannot be deslected, when further instances are added.

Enter the **Instance root directory** typically this will be a path such as E:\MSSQL\

Enter the **Shared feature directory** typically this will be a path such as E:\MSSQL\

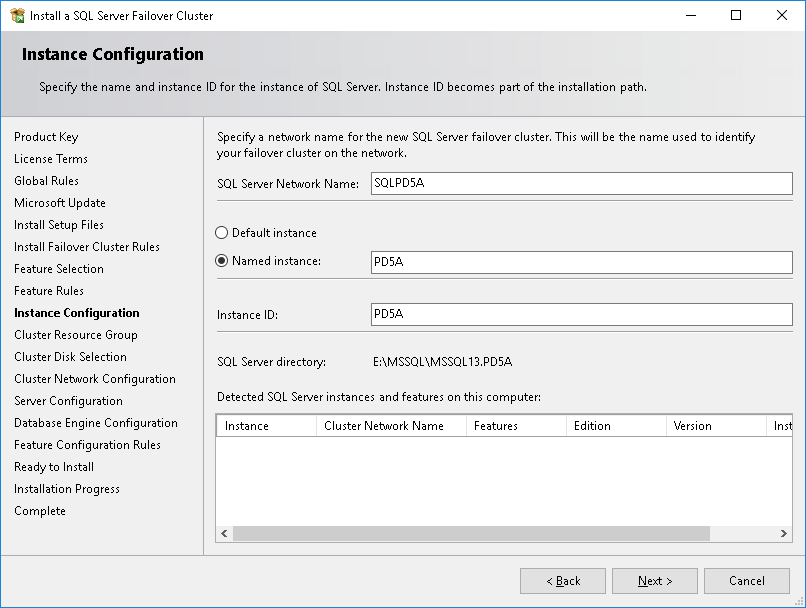
Enter the **Shared feature directory (x86)** typically this will be a path such as E:\MSSQL(x86)\

Note that Shared feature directory and Shared feature directory (x86) edit controls will be defaulted and un-editable when installing the second or subsequent instance in a multi-instance failover cluster configuration.

Click **Next**

Note that the **Feature Rules** page will not be shown if no issues are found. It is possible to navigate back to the **Feature Rules** page using the **< Back** button from the **Instance Configuration** page.

The **Instance Configuration** page is displayed



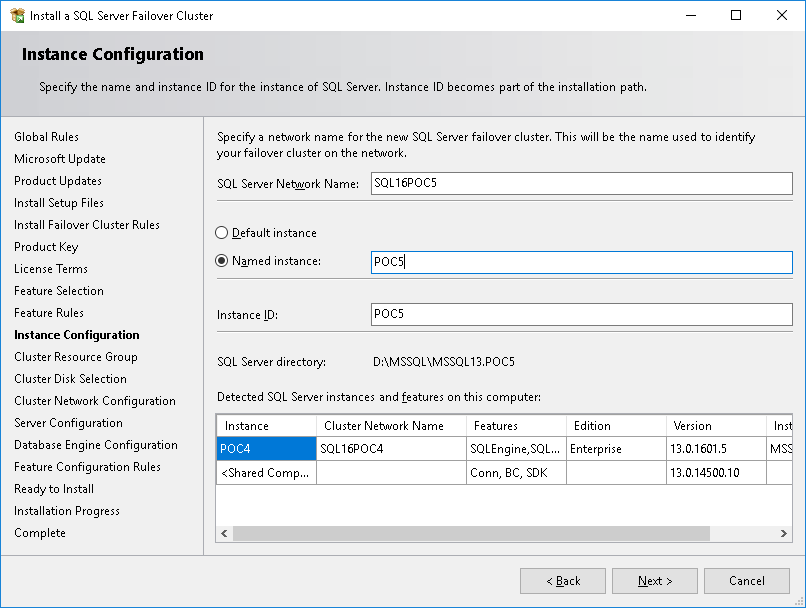
Enter the **SQL Server Network Name** for the cluster resource group.

If this is to be a default instance of SQL Server, ensure the **Default instance** radio button is selected.

If this is to be a named instance of SQL Server, ensure the **Named** **instance** radio button is selected and enter the instance name in the edit box to its right.

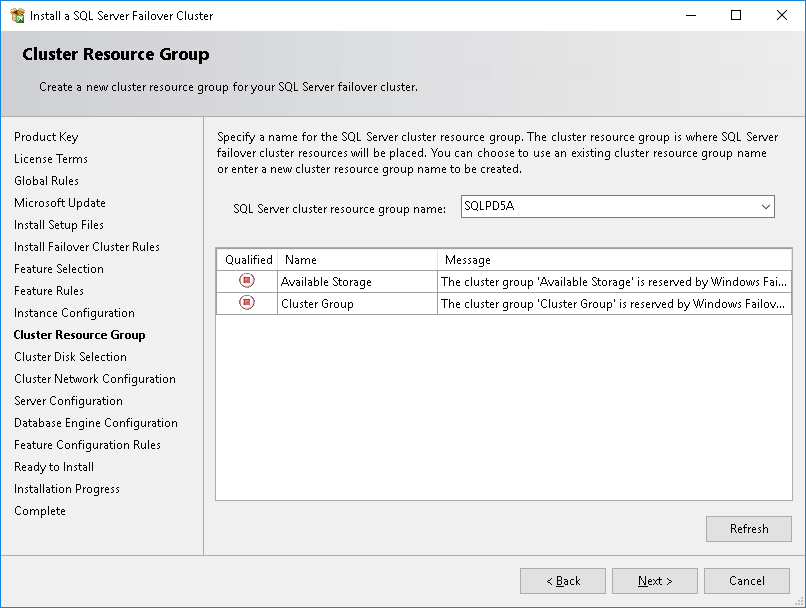
The **Instance ID** will default to MSSQLSERVER for a default instance or the instance name for a named instance, ensure this is correct before proceeding.

Note that when installing multiple SQL Server instances on the same cluster that any previously installed instances or shared components will be listed under **Detected SQL Server instances and features on the computer:**



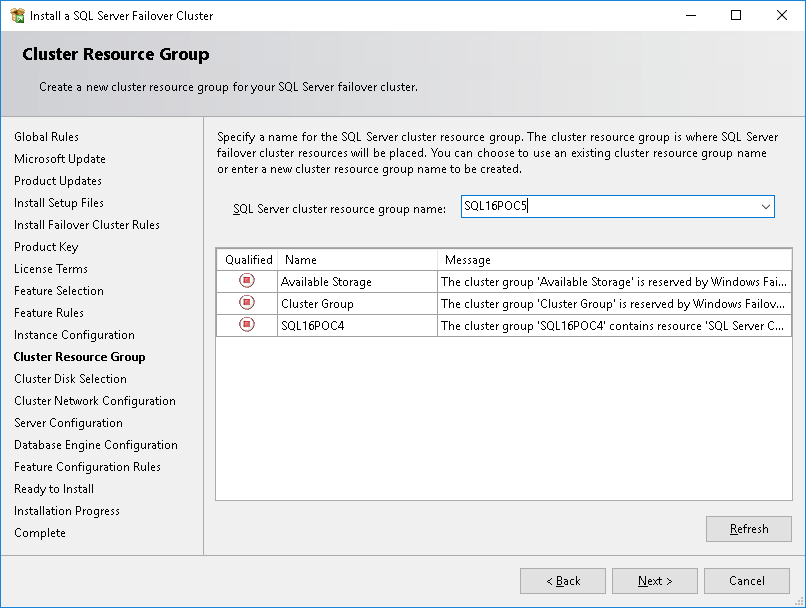
Click **Next**

The Cluster Resource Group page is displayed.

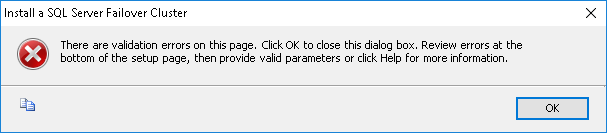


The SQL Server cluster resource group name will default to *SQL Server (<Instance Name>)* or *SQL Server (MSSQLSERVER)* for a default instance; change this to the **SQL Server Network Name** selected on the previous page.

Note that when installing multiple SQL Server instances within the same cluster and previously installed resource groups will be listed alongside the Available Storage and Cluster Group resources.



If you attempt to use the same name as an existing resource group and click **Next** validation will fail with the following popup:

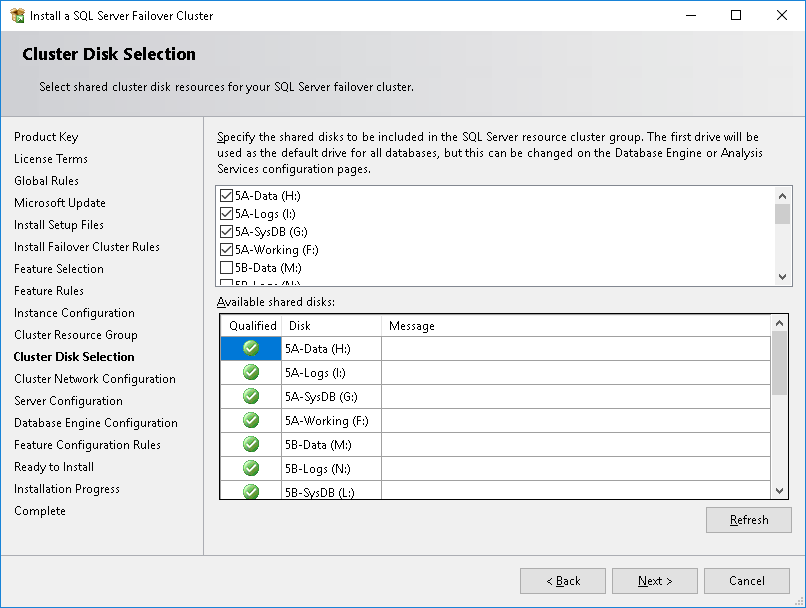


If this occurs, click **OK**. Additional information on validation errors will be shown at the bottom of the **Cluster Resource Group** page, for example:



Correct the error, then click **Next**

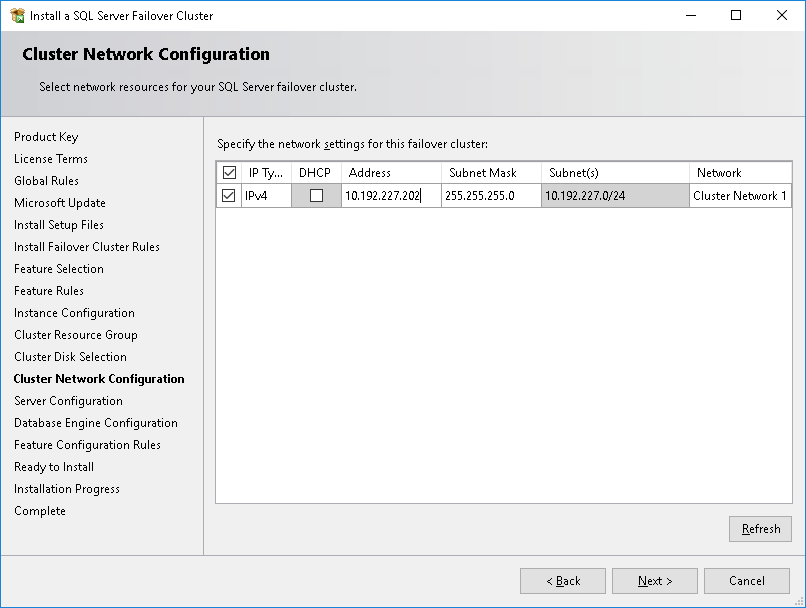
The Cluster Disk Selection page is displayed.



Select the disks from those available which are to be assigned to this SQL Server resource. In a multiple instance failover cluster remember that there may be disks allocated for additional SQL Server resource groups, ensure the correct disks are selected for the instance being installed.

Click **Next**

The **Cluster Network Configuration** page is displayed.

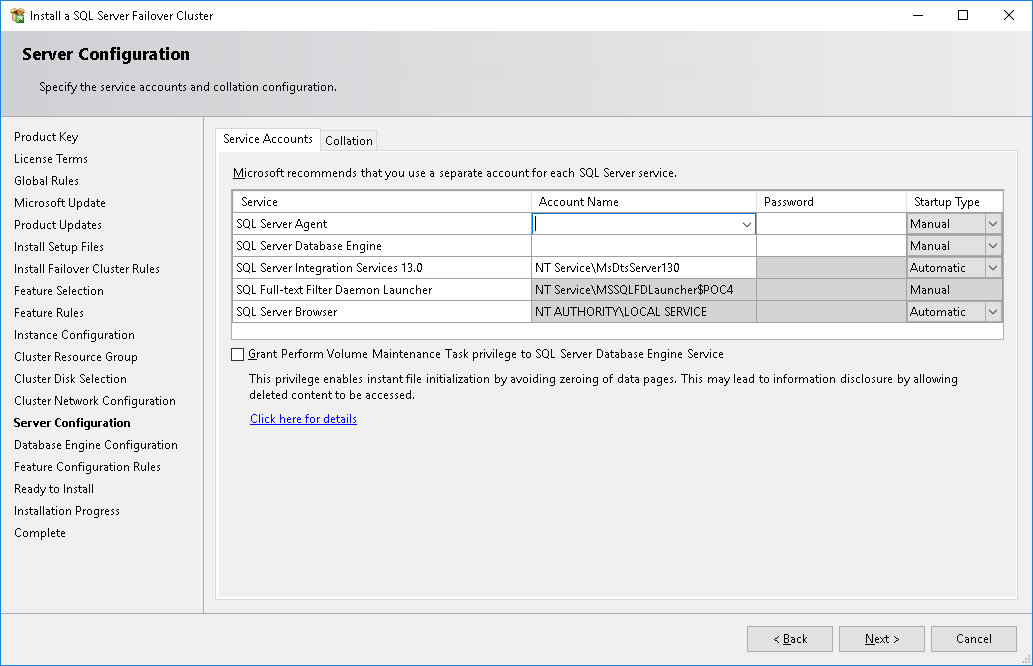


Check **IPv4** and enter the allocated **Address**, ensure **DHCP** is *not* checked.

If the cluster has multiple networks assigned, for example a public network and private heartbeat network, ensure the correct **Network** is selected.

Click **Next**

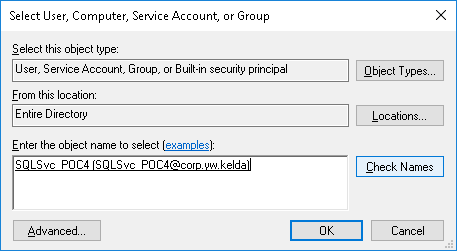
The **Server Configuration** page is displayed.



For a failover cluster installation, no default account is supplied under **Account Name** as virtual service accounts cannot be used, Windows domain accounts *must* be used for the SQL Server Database Engine and SQL Server Agent services.

The **Account Name** and **Password** will need to be updated (note that it is not possible to change the default account for the *SQL Server Browser* service from NT AUTHORITY\LOCAL SERVICE). To update the service accounts either type (or paste from clipboard) the account name in to the **Account Name** field using domain\account\_name format or, alternatively use the dropdown arrow to the right of the account name and select **<<Browse…>>**

Then use the standard Windows **Select User, Computer, Service Account, or Group** window to locate the service account.

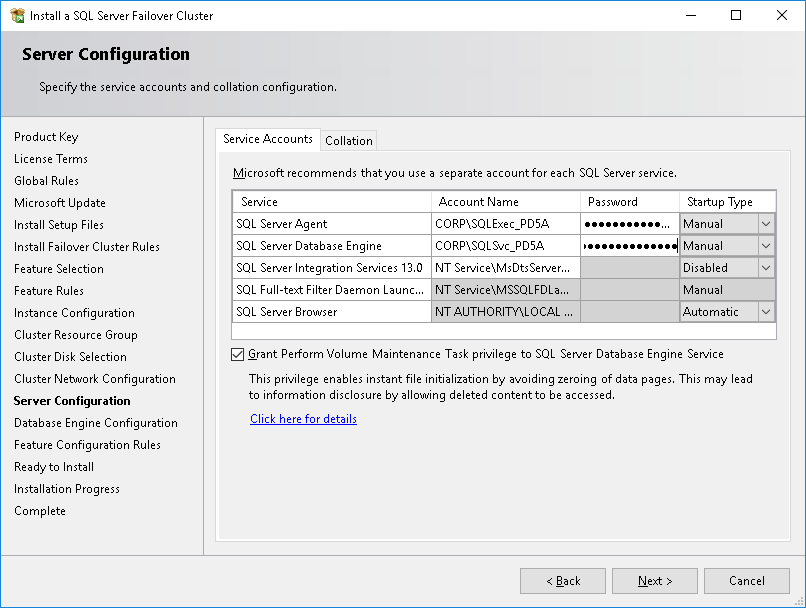


Note the services are listed in alphabetical order; the *SQL Server Agent* appears before the *SQL Server Database Engine*, check carefully to ensure that the correct accounts are used.

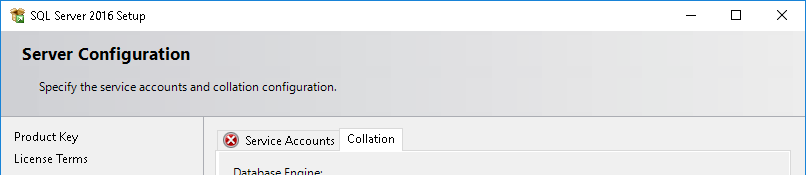
Once the **Account Name** has been specified enter the associated **Password**.

Note the **Startup Type** for *SQL Server Agent, SQL Server Database Engine* is Manual and cannot be changed; these services are managed by the Windows Failover Cluster service. If SQL Server Integration Services is being installed its **Startup Type** should also be set to **Disabled** in most cases.

Ensure **Grant Perform Volume Maintenance Task privilege to the SQL Server Database Engine Service** checkbox is checked.



Select the **Collation** tab. The account information entered is validated, if any errors are found an error symbol is displayed on the title of the **Service Accounts** tab as illustrated.



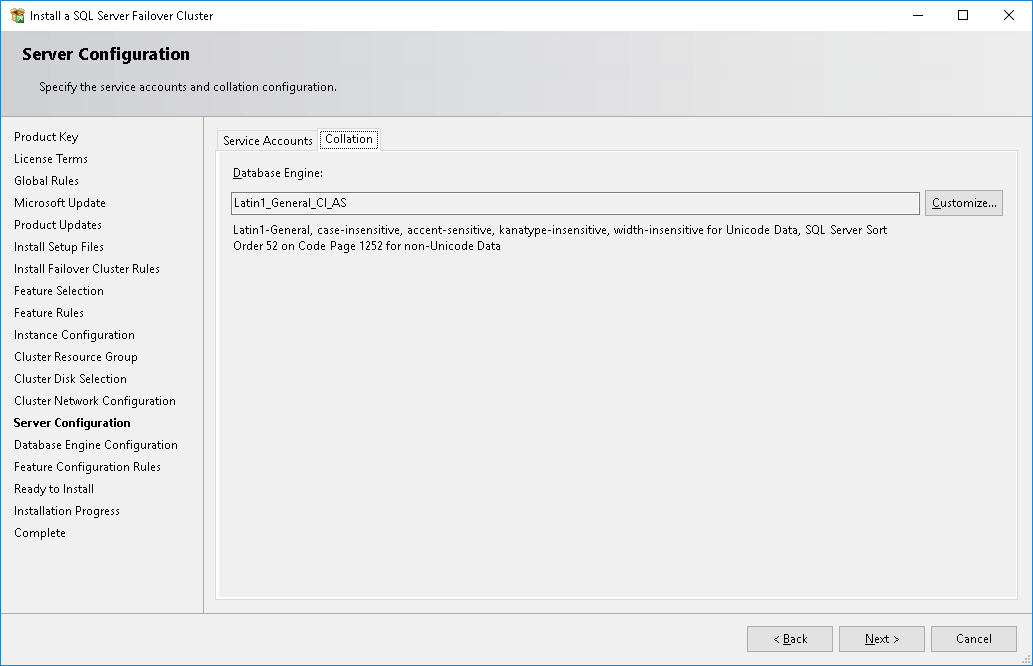
If this is the case return to the **Service Accounts** tab where more information is available.

The following errors may be the result of a mistyped account name, an incorrect password, the account being locked, or using an account name which has been created but which has not yet replicated to all domain controllers and therefore cannot be found by SQL Server Setup.



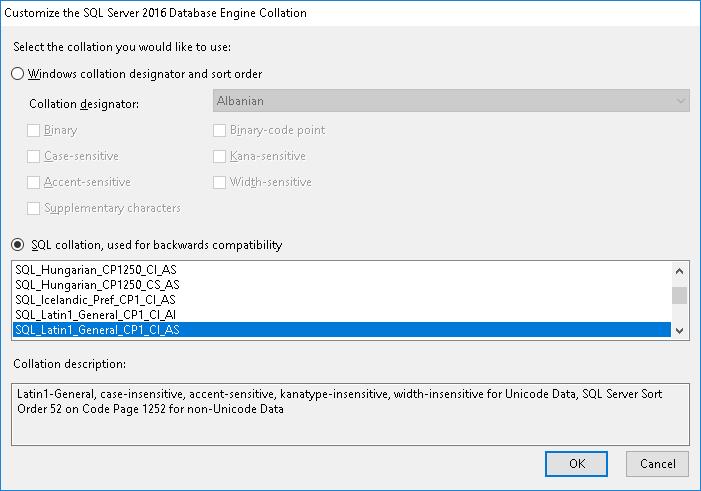


Resolve any errors and select the **Collation** tab



The **Collation** tab will display the default collation of the instance being installed, review this against the build specification. To change the default collation click **Customize…**

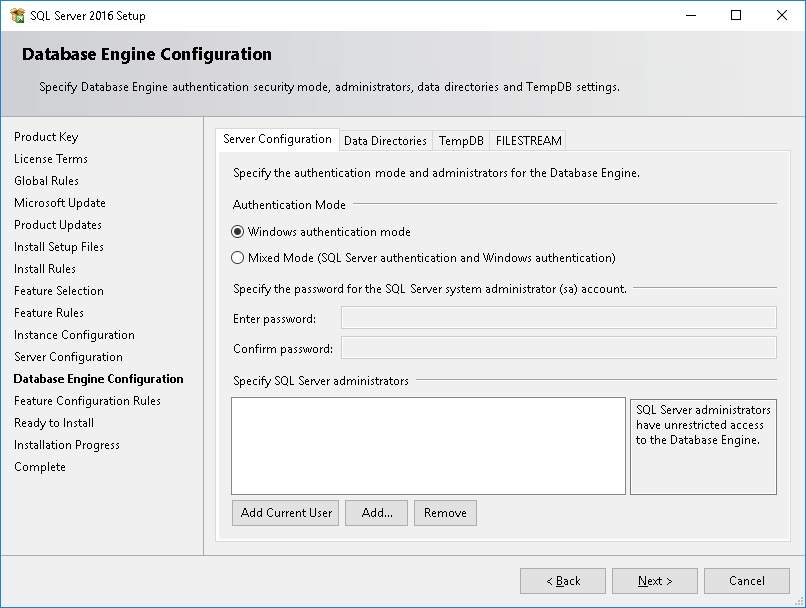
Then select the desired collation from the **Customize the SQL Server 2016 Database Engine Collation** dialog



Click **OK**

Once the correct collation is selected click **Next**

The **Database Engine Configuration** page is displayed



If the instance is to use Windows integrated security *only* ensure the **Windows authentication mode** is selected.

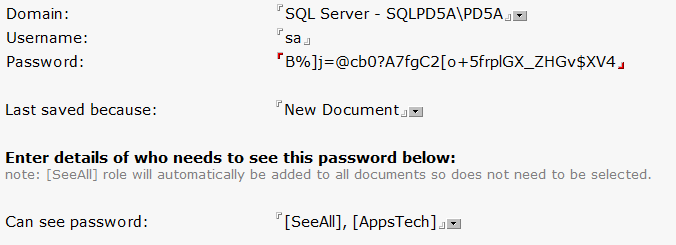
If both Windows and SQL Server authentication are to be used select **Mixed Mode (SQL Server authentication and Windows authentication)**.

If Mixed Mode security is selected the **Enter password** and **Confirm password** edit boxes under **Specify the password for the SQL Server system administrator (sa) account** are enabled, generate and enter a complex password in both.

Record this password in the IT Password Repository as:

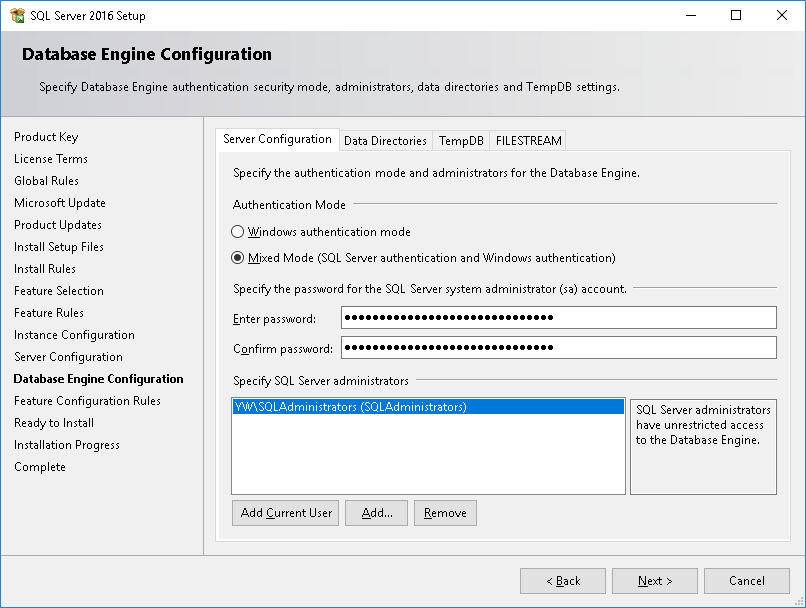
* Domain: SQL Server - <Instance Name>
* Username: sa
* Can see password: SeeAll, AppsTech

An example of this is shown below.



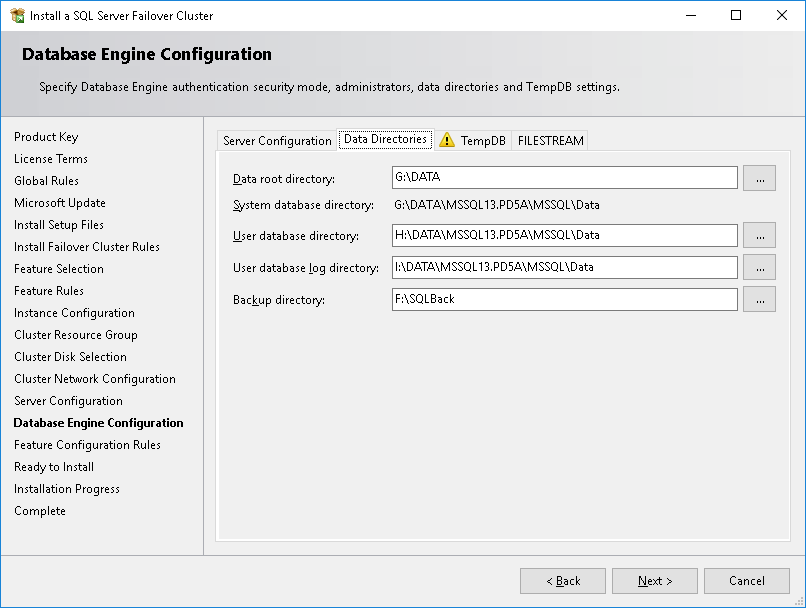
Ensure the sa password is recorded before proceeding with the installation.

Under the **Specify SQL Server administrators** list box click **Add…** and use the **Select Users, Computer, Service Accounts, or Groups** dialog to add the group(s) to be granted sysadmin access to the instance, YW*\SQLAdministrators* must be added along with application specific or development groups appropriate to the instance.



Select the **Data Directories** tab

The **Data root directory** will default to the first available cluster disk.



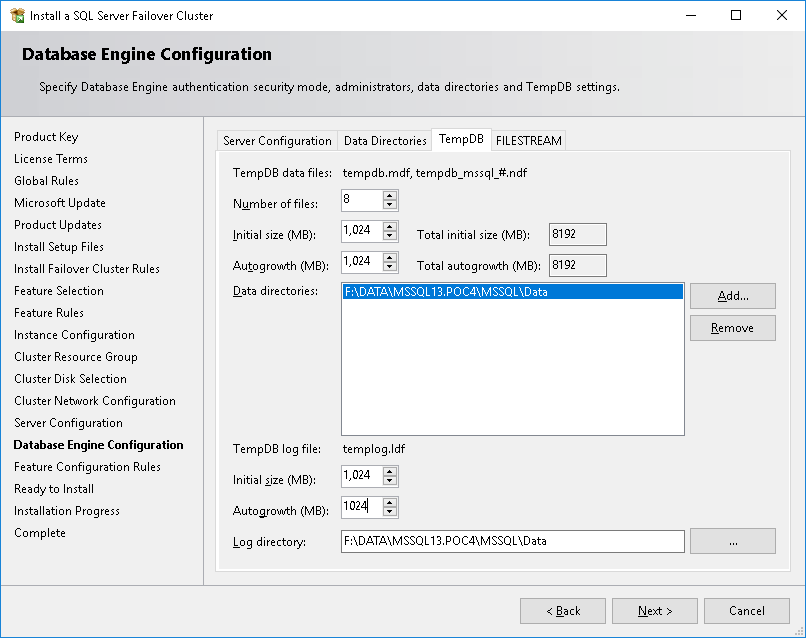
Referring to the build specification, determine the drive letter to be used for the *system databases*. Amend the **Data root directory** to <system database drive>:\DATA\. The **System database directory** will be updated to reflect changes made to **Data root directory**; note that the **System database directory** *cannot* be specified directly.

Change the drive letter, leaving the rest of the path unmodified, for the **User database directory** to the drive letter given in the build specification for user database data files.

Change the drive letter, leaving the rest of the path unmodified, for the **User database log directory** to the drive letter given in the build specification for user database transaction log files.

Change the **Backup directory** to <drive>:\SQLBack where <drive> is the drive letter given in the build specification for database backups. Note that typically TDP will be used for routine backups; if for some reason this is not the case and SQL Server native disk backups are being used consideration should be given to a separate disk for backup.

Once the correct paths have been entered and verified, select the **TempDB** tab.



The number of TempDB data files should be determined as being equal to the number of processor cores/vCPUs for 8 cores or fewer, 8 for more than 8 cores; adjust the **Number of files**, under **TempDB data files**, to this value (note that 8 data files is the maximum that can be specified at setup).

Specify the **Initial size (MB)**, under **TempDB data files**, for each tempdb data file, 1024 MB is recommended and the maximum that can be specified at setup. Similarly, specify the **Autogrowth (MB)** for the datafile(s); 1024 MB is recommended and the maximum which can be specified at setup.

Under **TempDB log file** specify the **Initial size (MB)** for the tempdb transaction log– 1024 MB is recommended and the maximum that can be specified at setup.

Also under specify the **Autogrowth (MB)** for the tempdb transaction log file1024 MB is recommended and the maximum that can be specified at setup.

The **Data directories** list will contain a single value which is the **System database directory** from the Data Directories tab; this is also the default value of the **Log directory** for tempdb.

If tempdb is to be located on a dedicated disk, as recommended, these paths will need amending.

Select the **Log directory** and change the drive letter of the path displayed to the drive letter assigned to the tempdb disk, do not make any other changes.

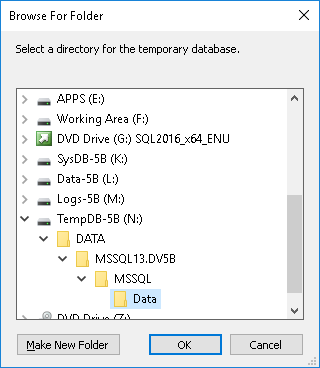
Then select the full text of the **Log directory** value and copy this to the clipboard.

Open a command prompt as Administrator and enter md *path* where path is the value just copied.

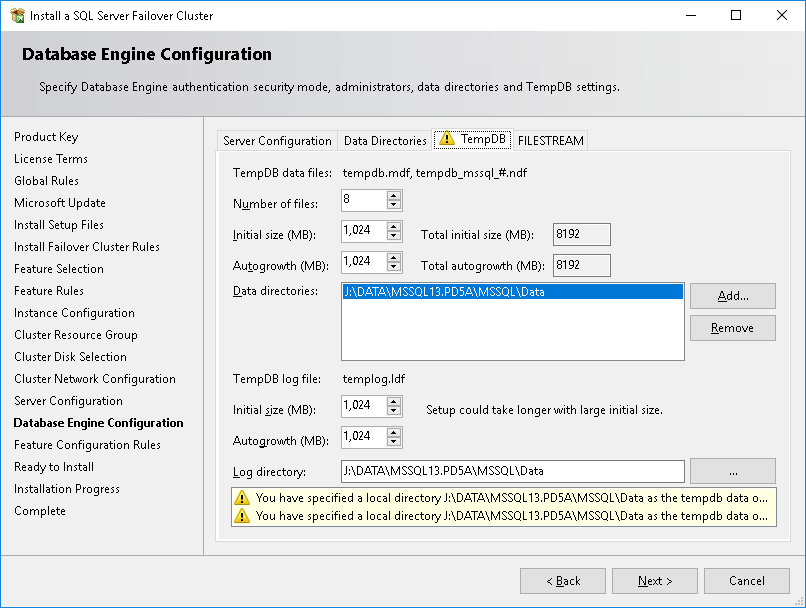
Then select the single entry in the **Data directories** list box and click **Remove**.

Click **Add**.

Use the **Browse For Folder** dialog to navigate to the folder just created.



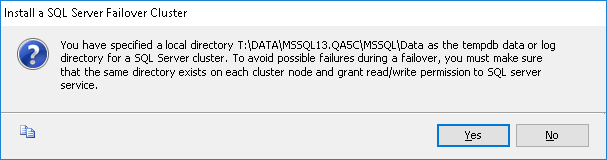
Click **OK**



If a locally attached disk is being used for TempDB rather than SAN-attached storage (see Disk Layout Considerations) setup will warn you that you must ensure that the same drive and path must exist on each cluster node.

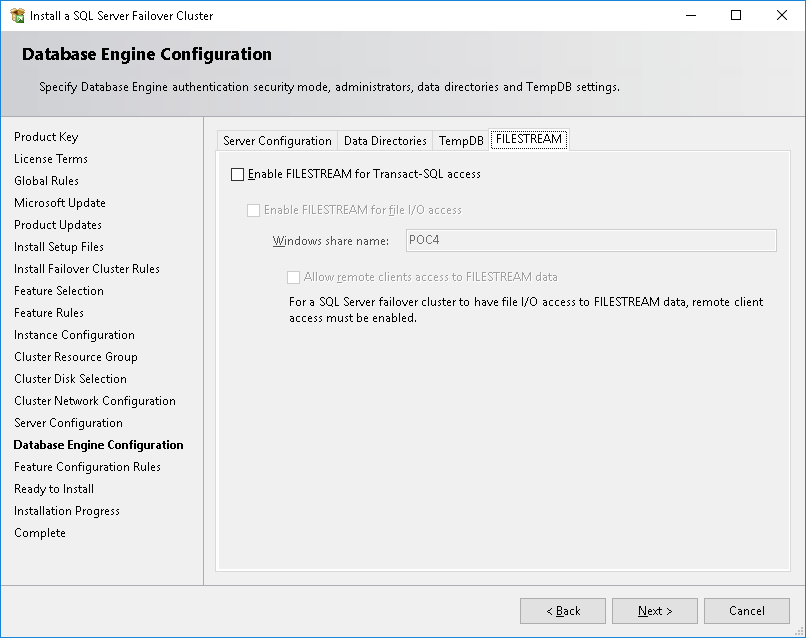
This is illustrated above by the text with yellow warning triangles and the addition of a warning triangle to the **TempDB** tab.

When you navigate away from the **TempDB** tab the following will be displayed.



Click **Yes**

Select the **FILESTREAM** tab.

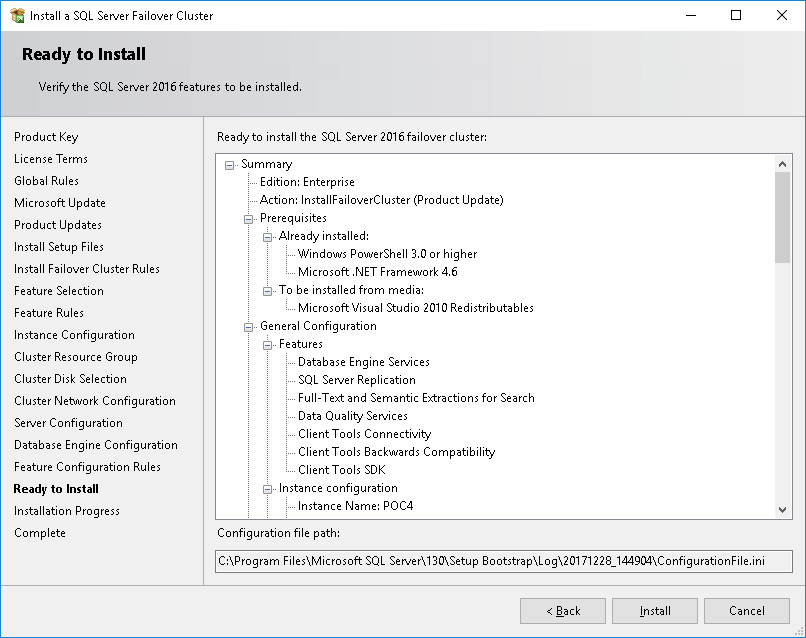


The configuration of FILESTREAM is not part of the standard build; **Enable FILESTREAM for Transact-SQL access** should be left unchecked. If FILESTREAM is required for an application reference should be made to

Click **Next**

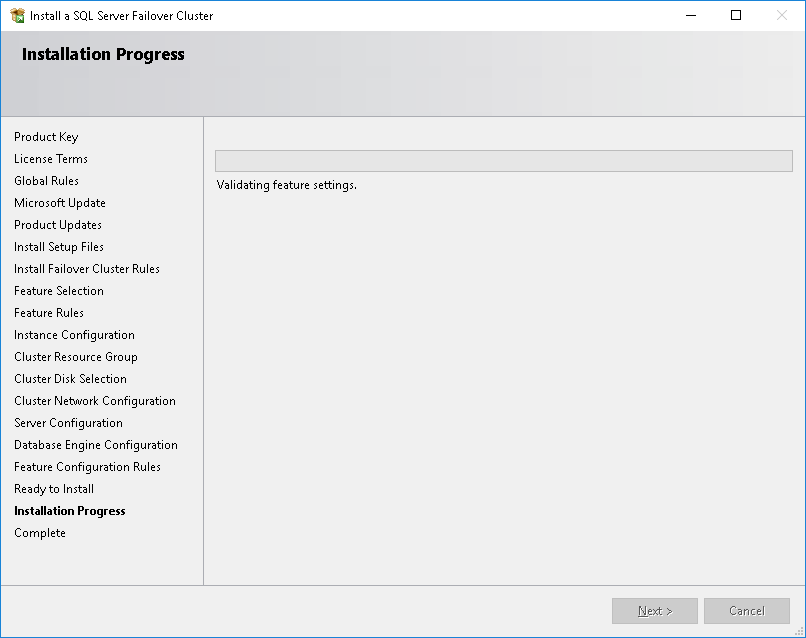
Note that the **Feature Configuration Rules** page will not be shown if no issues are found. It is possible to navigate back to the **Feature Configuration Rules** page using the **< Back** button from the **Ready to Install** page.

The Ready to Install page is displayed

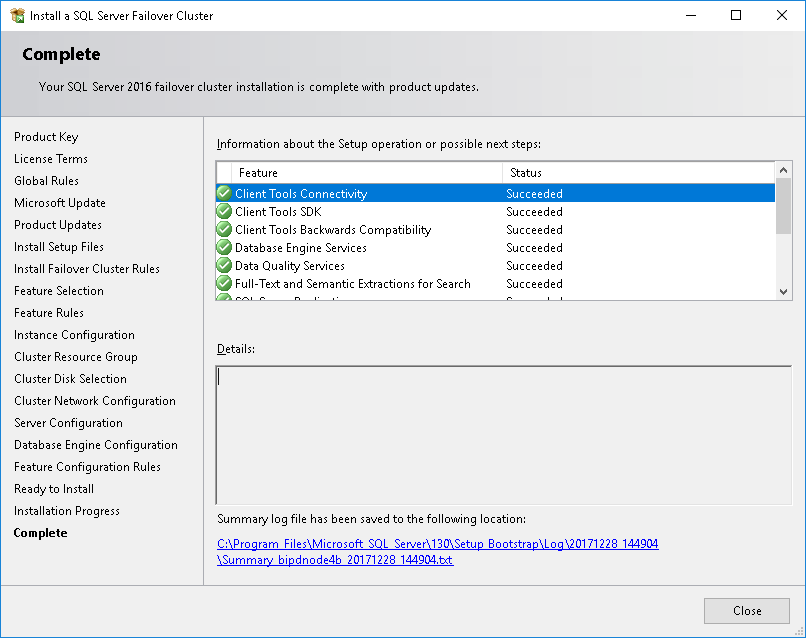


Review the configuration, if any errors are found use the **< Back** button to return to the appropriate page and resolve them. If the configuration is correct, click **Install**

The **Installation Progress** page is displayed



No action, other than **Cancel** the installation is possible; this page is only provided to monitor installation progress. Once setup is complete, the **Complete** page is displayed.



Review the list of features installed and ensure all show a **Status** of Succeeded.

Click Close

Setup may require a reboot to complete installation, if this is required setup will prompt you to reboot. It is recommended that you choose to reboot later, once trace flags have been configured.

For a multi-instance failover cluster repeat the above for each additional instance, then proceed to Configure Trace Flags. For a single-instance failover cluster, proceed to Configure Trace Flags.

## Add Additional Nodes to the Cluster

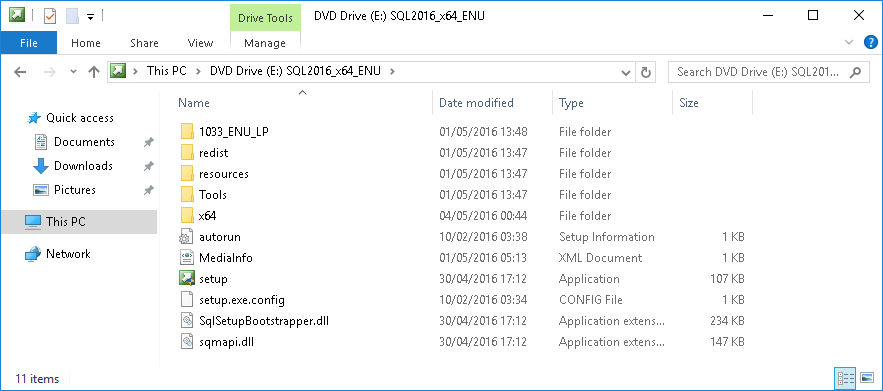
The instructions below will need to be carried out on each additional node of the cluster in turn and repeated for each instance in a multi-instance failover cluster configuration.

Log on to the cluster node with Administrative rights.

### Mount the SQL Server installation ISO image

On the first cluster node, mount the appropriate SQL Server 2016 ISO:

* For MSDN licenced installations media are located at \\SQLDBA\Media$\SQL Server 2016\MSDN
* For Production licenced installations media are located at \\SQLDBA\Media$\SQL Server 2016\Volume Licence



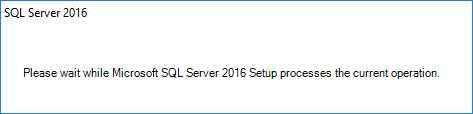
### Setup SQL Server 2016

Launch Setup.exe using the right-mouse context menu and select “**Run as administrator**”

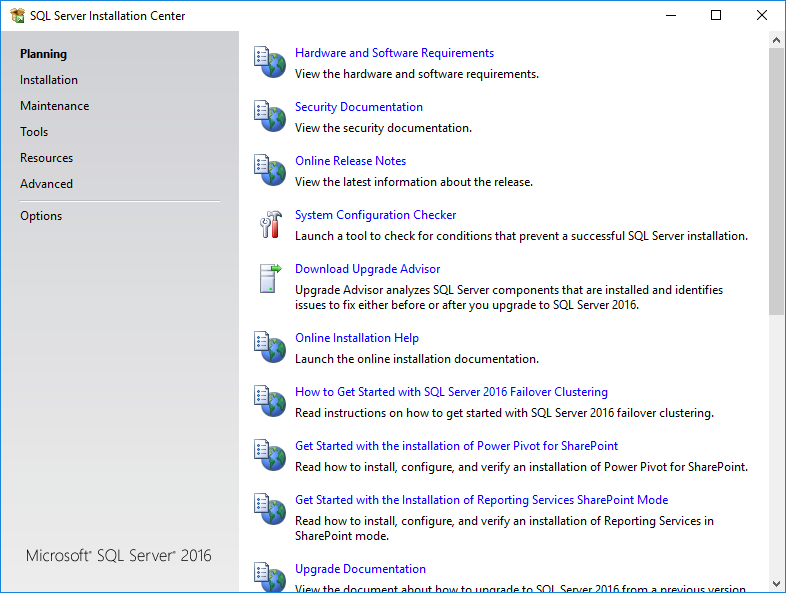
If prompted by User Account Control “Do you want allow this app to make changes to the device?” (as illustrated below), click **Yes**



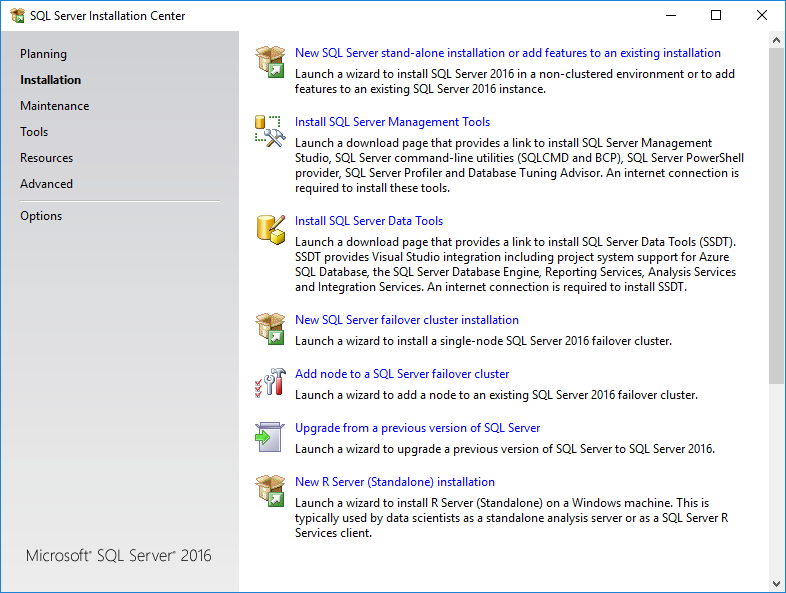
The setup will launch and display the following message:



After a few seconds the **SQL Server Installation Center** will be displayed:

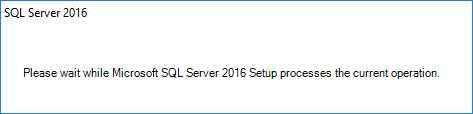


Select Installation from the left-hand side of the SQL Server Installation Center

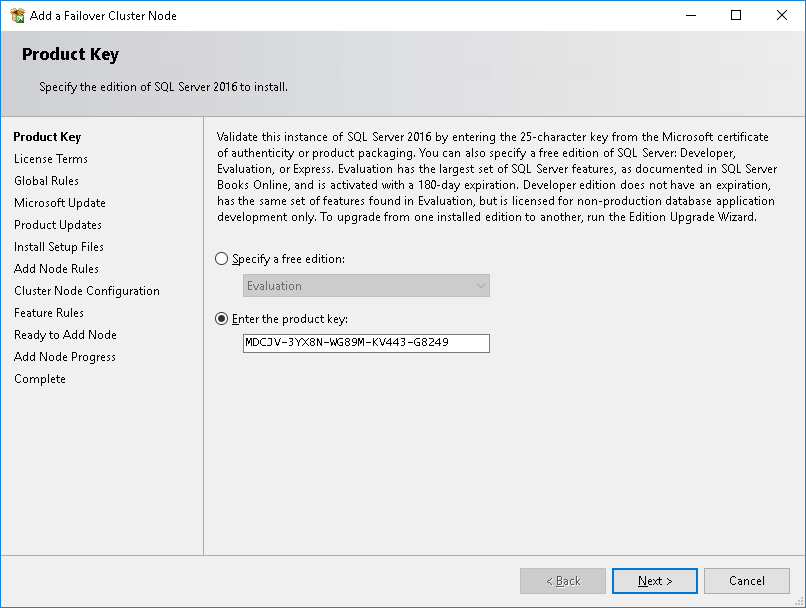


Click **Add node to a SQL Server failover cluster**

The following message will be displayed for a few seconds whilst the setup is loading

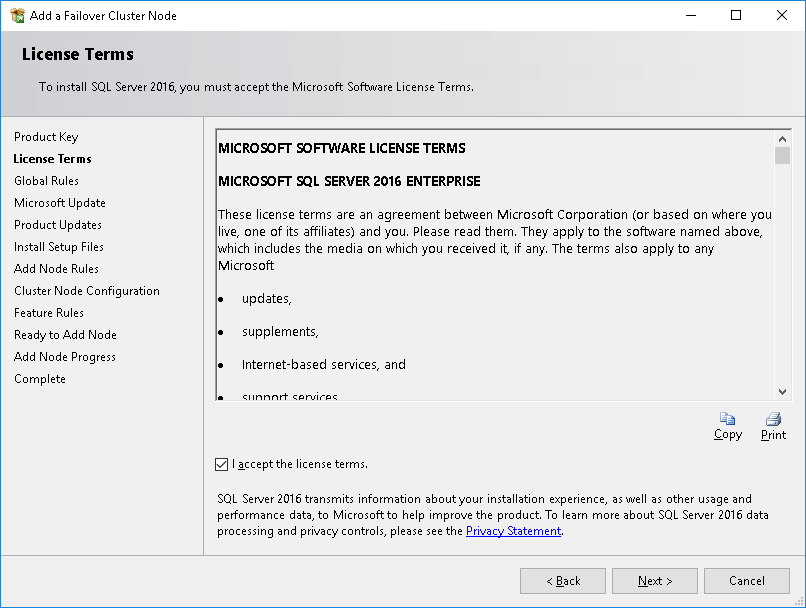


The **Add a Failover Cluster Node** wizard will be displayed at the **Product Key** page.



The same product key must be used on each node of the cluster.

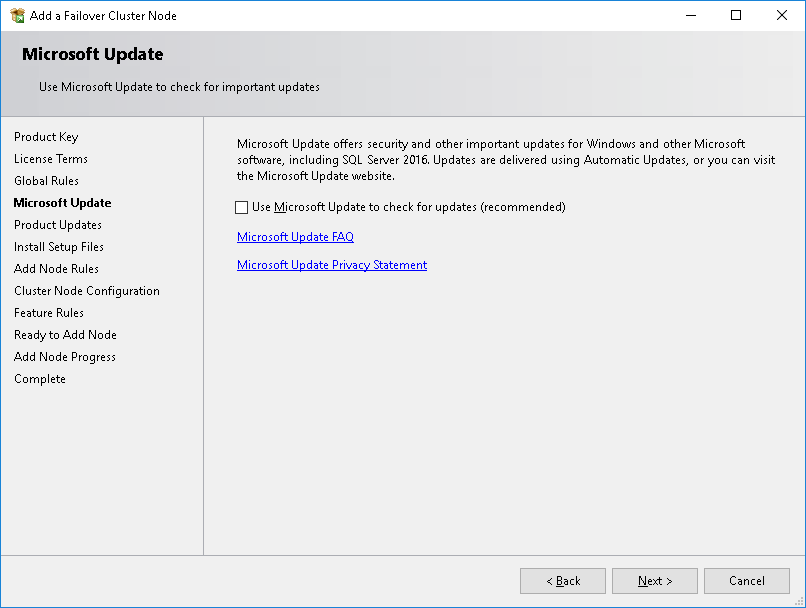
Click **Next**



Check “I accept the license terms” and click Next

The **Global Rules** page will be displayed.

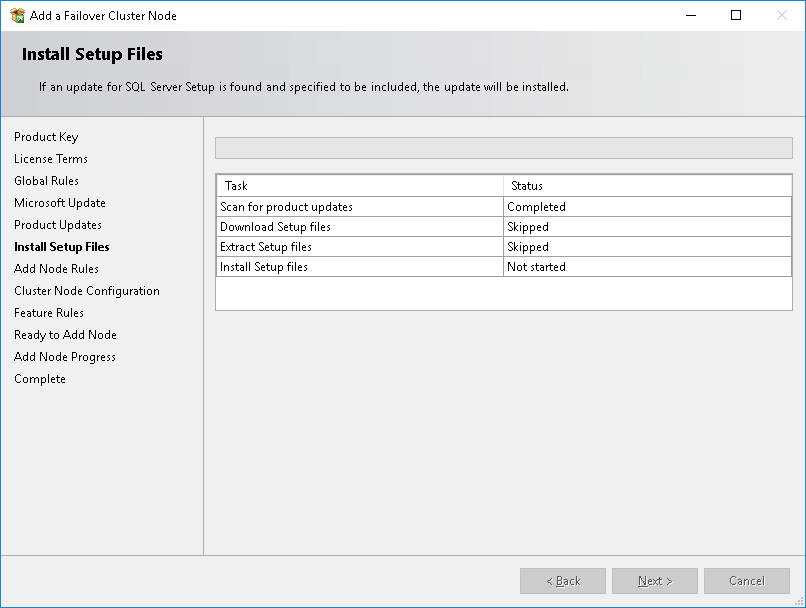
If all rules are passed the wizard will automatically display the **Microsoft Update** page, you may use **Back** to display the **Global Rules** page and review the rule evaluation.



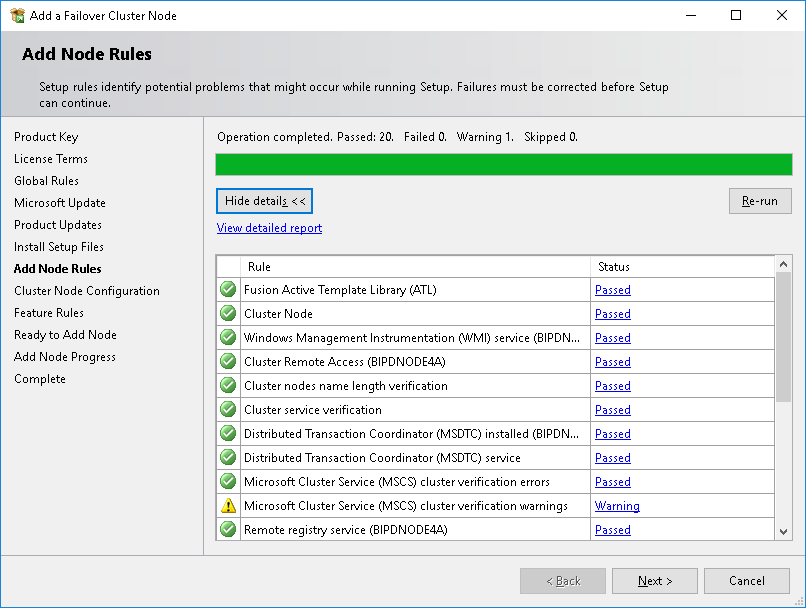
Leave **Use Microsoft Update to check for updated (recommended)** unchecked, updates will be manually applied to allow control of the version of SQL Server built. *This skips the Product Updates page.*

Click **Next**

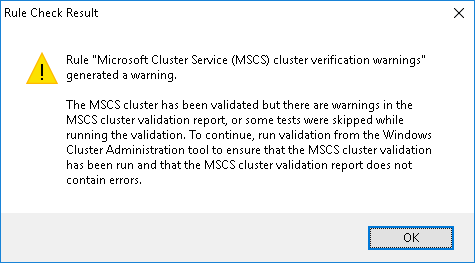
The **Install Setup Files** page will be displayed.



Once setup file installation is completed the **Add Node Rules** page is displayed.



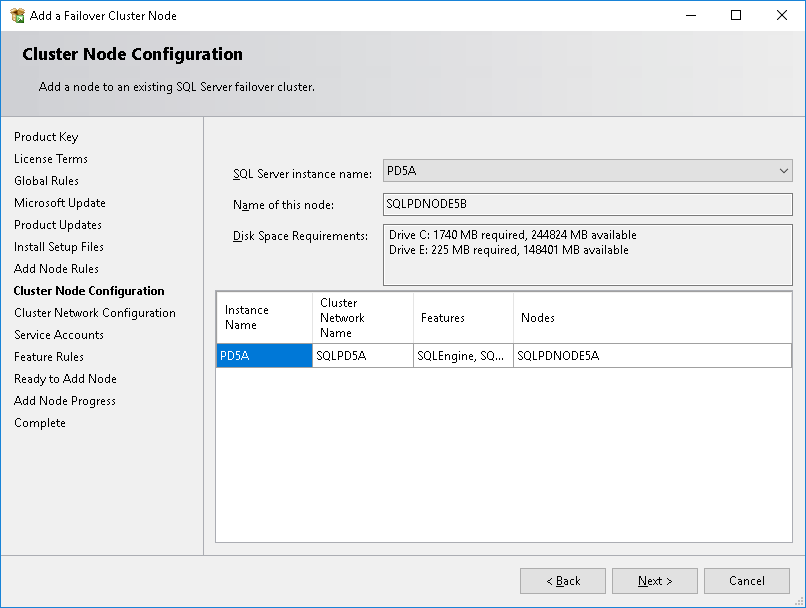
You may also see a Warning for Microsoft Cluster Service (MSCS) cluster verification warnings



Refer to the earlier sections on Cluster Validation

Once **Add Node Rules** have been evaluated review the findings and resolve any failures, then click **Next**

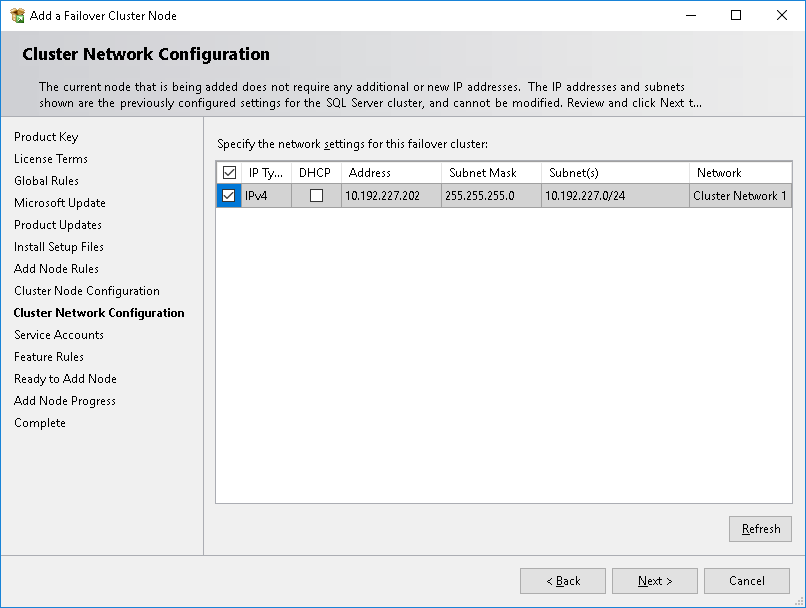
The Cluster Node Configuration is displayed



Select the **SQL Server instance name** from the drop-down list, the selected instance will be installed on the node. For a multiple instance failover cluster, the setup must be run once for each instance.

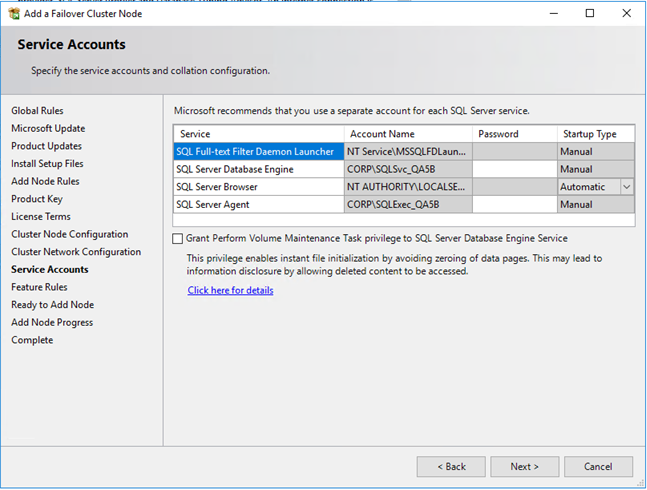
Note, the **SQL Server instance name** will only list those instances which have *not* been installed on this node.

The Cluster Network Configuration page is displayed



Click **Next**

The **Service Accounts** page is displayed.



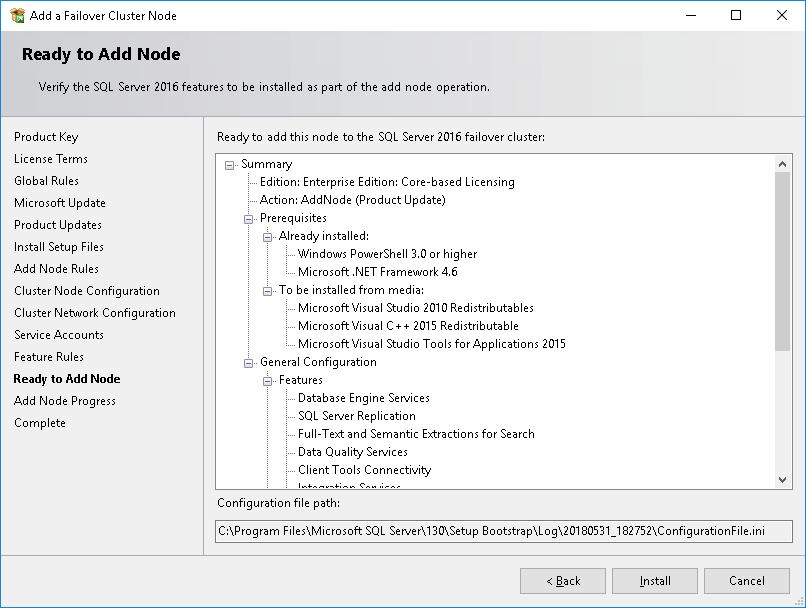
The **Account Name** will be pre-populated with the accounts selected when installing the instance on the first node, enter the appropriate service account **Passwords**

Ensure Grant Perform Volume Maintenance Task privilege to the SQL Server Database Engine Service checkbox is checked.

If Integration Services is being installed change the **Startup Type** of *SQL Server Integration Services 13.0* to **Disabled**

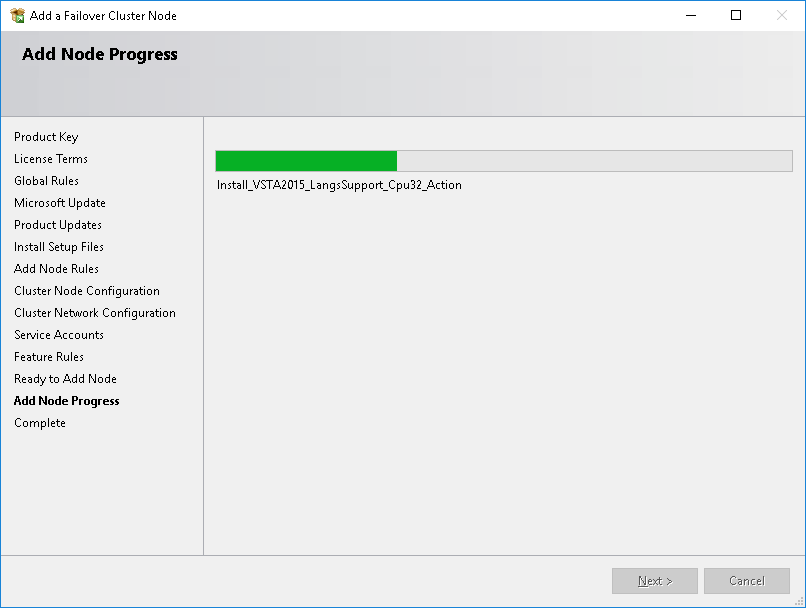
Click **Next**

The **Feature Rules** will be evaluated; if all rules are passed the **Ready to Add Node** page is displayed.

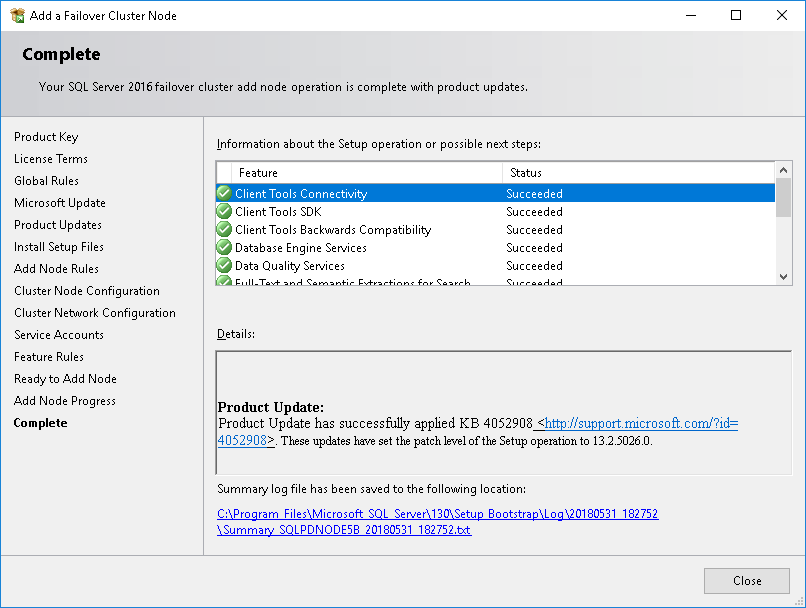


Review the configuration, then click **Install**

The **Add Node Progress** page will be displayed.



Once setup is finished the **Complete** page is displayed.



Click **Close**

Repeat the above process for any additional SQL Server instances in a multi-instance failover cluster and for all additional cluster nodes.

Once all instances have been installed on a cluster node configure the trace flags and error reporting, see above.

## Post Installation Tasks

It is recommended that the tasks “*Enable the Dedicated Admin Connection*”, “*Configure SQL Server TCP Ports*”, and “*Configure Trace Flags*” are completed on the active node for the instance and the instance failed to each node in turn to ensure registry settings replicate.

The task “*Configure SQL Server Error and Usage Reporting*” must be completed on each node of the cluster.

The task “*Remove the SQL Server CEIP Role*” is only performed from one cluster node.

### Enable the Dedicated Admin Connection

The Dedicated Admin Connection (DAC) is disabled by default, it will be enabled if the YW SQL Build is installed. However, if static TCP/IP ports are to be configured it is necessary to enable the DAC first so that its port can be configured.

To enable the DAC, connect to the instance using *SQL Server Management Studio* and execute the code below, if *SQL Server Management Studio* has not yet been installed a remote connection from your administrative PC or another server can be used. You must be e member of the sysadmin role on the target instance.

EXEC sp\_configure 'remote admin connections', 1;

GO

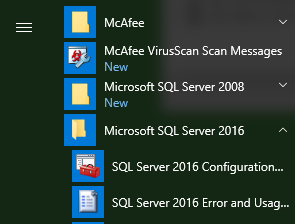
RECONFIGURE

GO

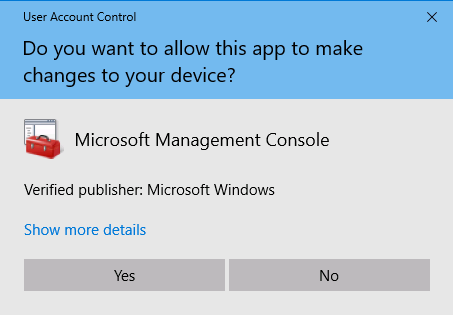
### Configure SQL Server TCP Ports

If specific ports are to be configured for SQL Server…

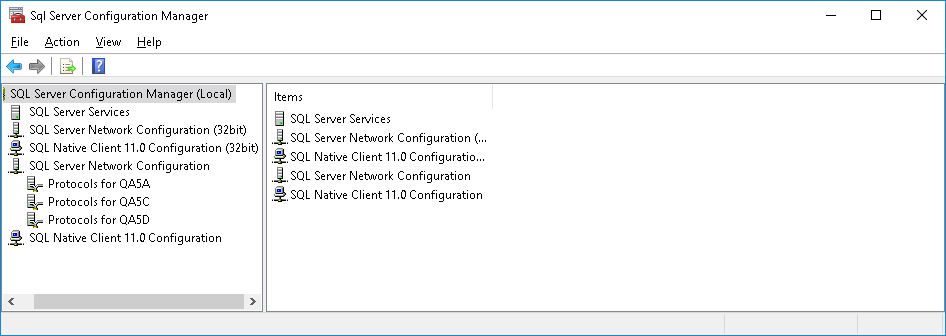
Start the **SQL Server Configuration Manager** from the Windows Start menu



If prompted by User Account Control “*Do you want allow this app to make changes to the device?*” (as illustrated below), click **Yes**



The **SQL Server Configuration Manager** will launch.

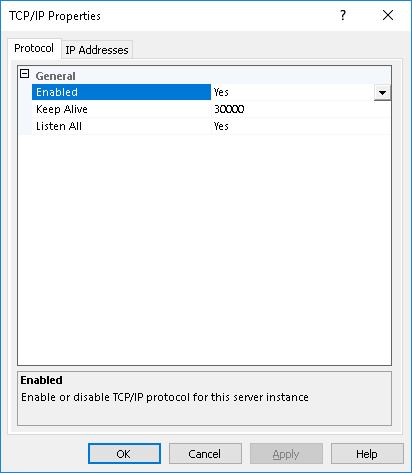


Navigate to **SQL Server Network Configuration**

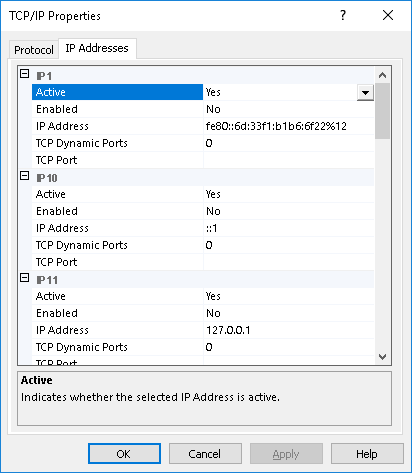
Select **Protocols for <Instance ID>** where instance ID is the identifier for a named instance or MSSQLSERVER for a default instance.

In the right-hand panel right-click **TCP/IP** andthen choose **Properties** from the context menu.

The **TCP/IP Properties** dialog is displayed.



Select the **IP Addresses** tab

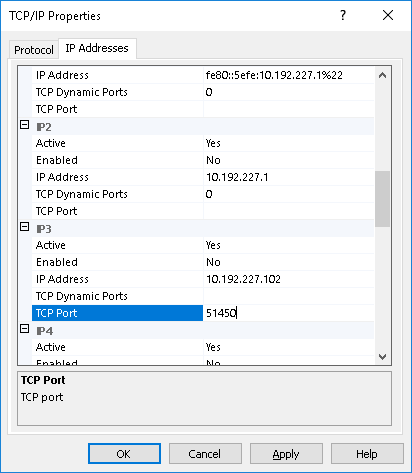


In the **TCP/IP Properties** dialog box several IP addresses appear in the format **IP1**, **IP2**, up to **IPAll**. One of these is for the IP address of the loopback adapter, 127.0.0.1. Additional IP addresses appear for each IP Address on the computer, (you will probably see both IP version 4 and IP version 6 addresses.)

Identify the IP address on which the SQL Server instance is configured to listen (this is **IP3** in the screenshot below)

If the **TCP Dynamic Ports** edit box contains 0 (or any other value), indicating the Database Engine is listening on dynamic ports, delete the value.

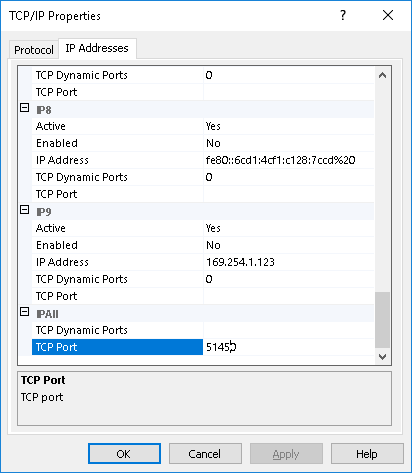
Enter the desired static port number in the **TCP Port** edit box (port 51450 has been selected in the screenshot below).



The process must be repeated for the **IPAll** address, select the **IPAll** IP address.

If the **TCP Dynamic Ports** edit box contains 0 (or any other value), indicating the Database Engine is listening on dynamic ports, delete the value.

Enter the desired static port number in the **TCP Port** edit box (port 51450 has been selected in the screenshot below).



The Dedicated Admin Connection port cannot be specified via the **SQL Server Configuration Manager** user interface so must be configured in the registry.

Open an Administrative Command Prompt

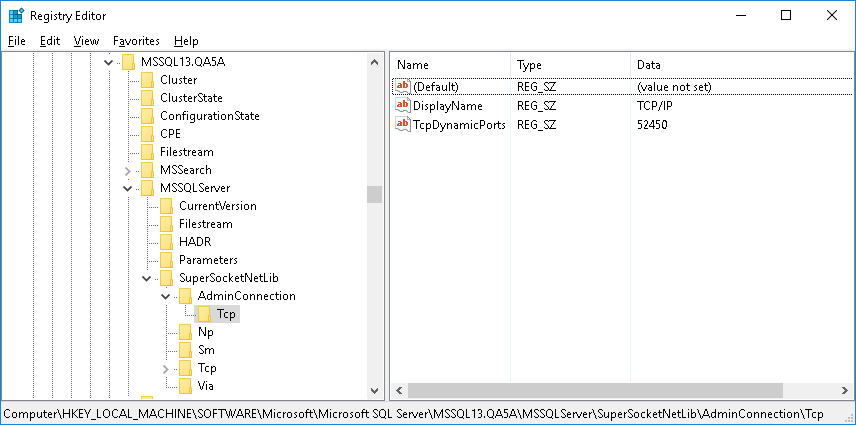
Enter regedit.exe

Navigate to the following registry key

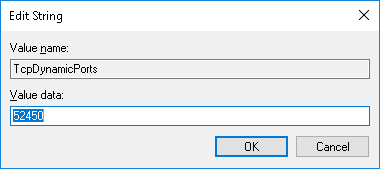
HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Microsoft SQL Server\MSSQL13.<Instance ID>\MSSQLServer\SuperSocketNetLib\AdminConnection\Tcp

Replacing <instance ID> with the appropriate instance identifier, for example

HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Microsoft SQL Server\MSSQL13.QA5A\MSSQLServer\SuperSocketNetLib\AdminConnection\Tcp



Double click the **TcpDynamicPorts** value, the **Edit String** dialog will be displayed.



Enter the desired static port in the **Value data**

Click **OK**

Alternatively, the port may be configured from the administrative command prompt without using the **Registry Editor** GUI with the command:

reg.exe ADD "HKLM\SOFTWARE\Microsoft\Microsoft SQL Server\MSSQL13.<instance\_ID>\MSSQLServer\SuperSocketNetLib\AdminConnection\Tcp" /v TcpDynamicPorts /t REG\_SZ /d <port> /f

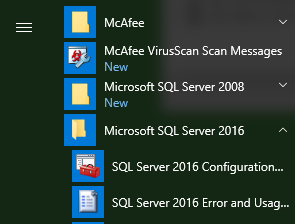
Replacing <instance ID> with the appropriate instance identifier and <port> with the static port number, for example:

reg.exe ADD "HKLM\SOFTWARE\Microsoft\Microsoft SQL Server\MSSQL13.QA5A\MSSQLServer\SuperSocketNetLib\AdminConnection\Tcp" /v TcpDynamicPorts /t REG\_SZ /d 52450 /f

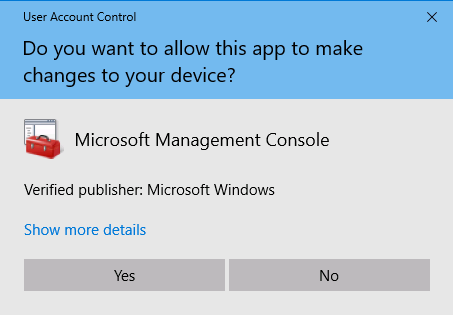
### Configure Trace Flags

This process must be repeated for each installed SQL Server instance on each node of the Windows Server Failover Cluster.

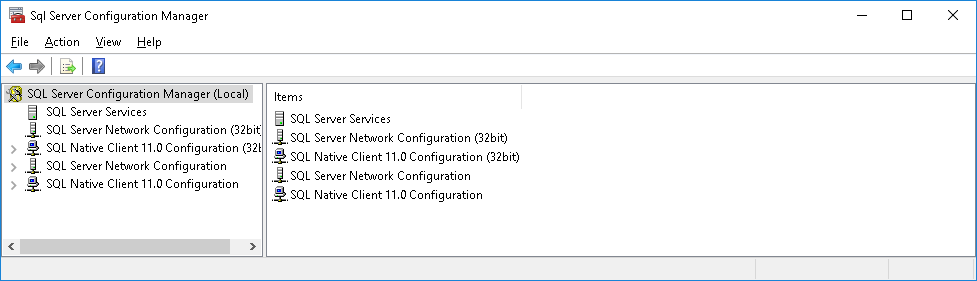
Start the **SQL Server Configuration Manager** from the Windows Start menu.



If prompted by User Account Control “Do you want allow this app to make changes to the device?” (as illustrated below), click **Yes**

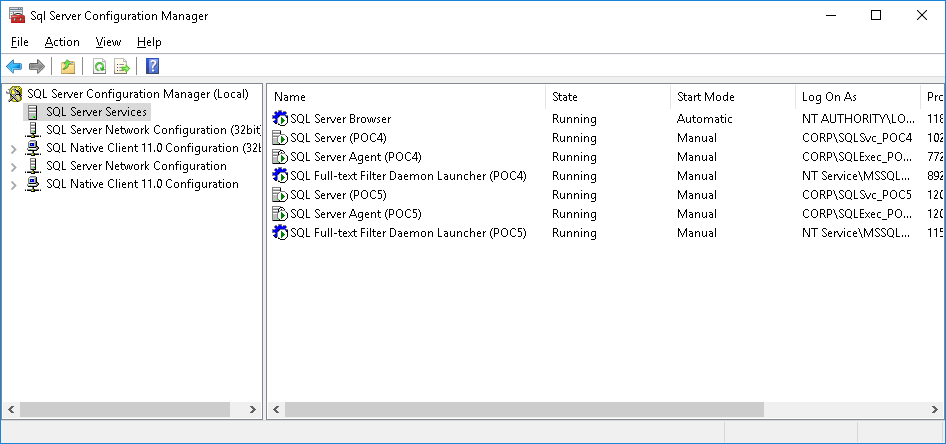


The **SQL Server Configuration Manager** will launch.

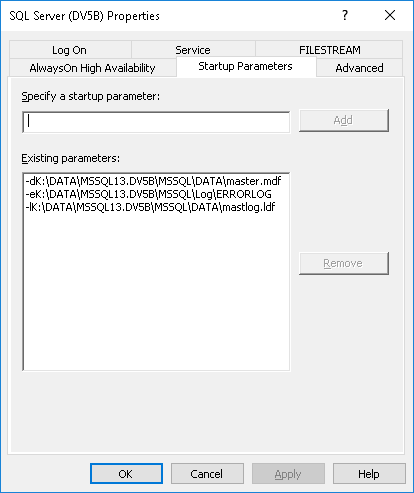


Navigate to **SQL Server Services** in the left-hand tree and select the SQL Server service which is then listed in the rght-hand panel, for a named instance this will be listed as **SQL Server (<Instance Name>)**, for a default instance the SQL Server service will appear as **SQL Server (MSSQLSERVER)**.

Right-click on the SQL Server service and select **Properties**



The instance properties dialog will be displayed; select the **Startup Parameters** tab.

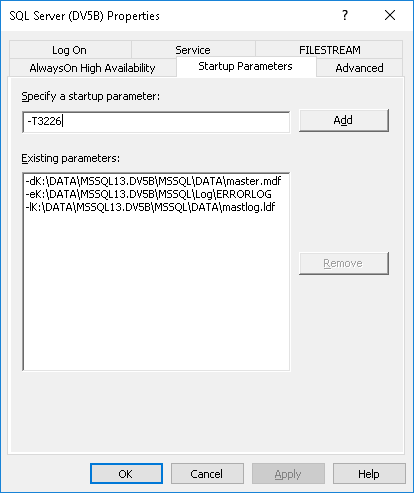


Review the list of trace flags below and determine which are to be applied – those listed here are *recommended* for all SQL Server 2016 installations unless a third-party application imposes a contrary requirement; additional trace flags may be required for specific applications and the vendor documentation should be consulted.

|  |  |
| --- | --- |
| Trace Flag | Purpose |
| -T3226 | By default, every successful backup operation adds an entry in the SQL Server error log and in the system event log. If you create very frequent log backups, these success messages accumulate quickly, resulting in huge error logs in which finding other messages is problematic.  With this trace flag, you can suppress these log entries. |

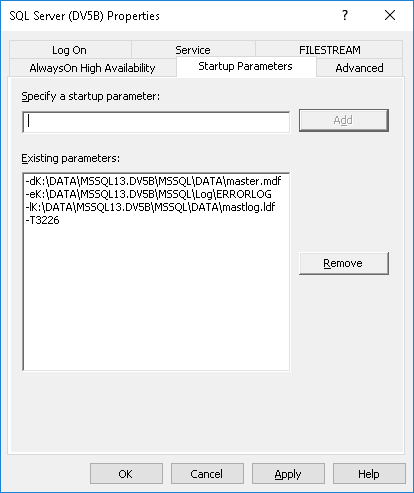
For each trace flag to be added enter the Trace Flag text the edit box **Specify a startup parameter**.

**\*\* Important \*\*** When specifying a trace flag with the -T option, use an uppercase "T" to pass the trace flag number. A lowercase "t" is accepted by SQL Server, but this sets other internal trace flags that are required only by SQL Server support engineers.



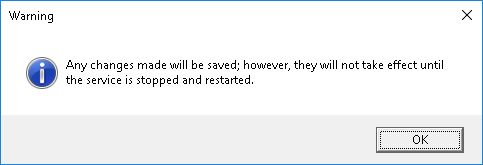
Click **Add**.

The trace flag will be added to the list of **Exisiting parameters**.



Once all required trace flags have been added click **OK**.

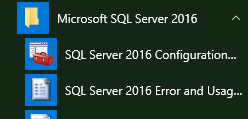
A Warning message will be displayed, click OK.



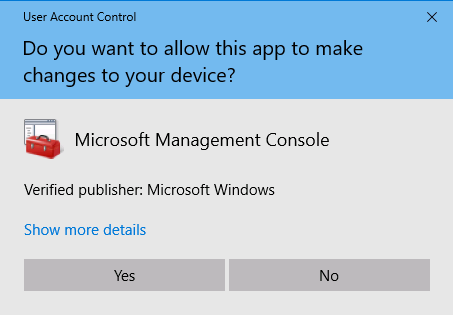
Reminder, in a multi-instance failover cluster repeat the above for each SQL Server instance installed.

### Configure SQL Server Error and Usage Reporting

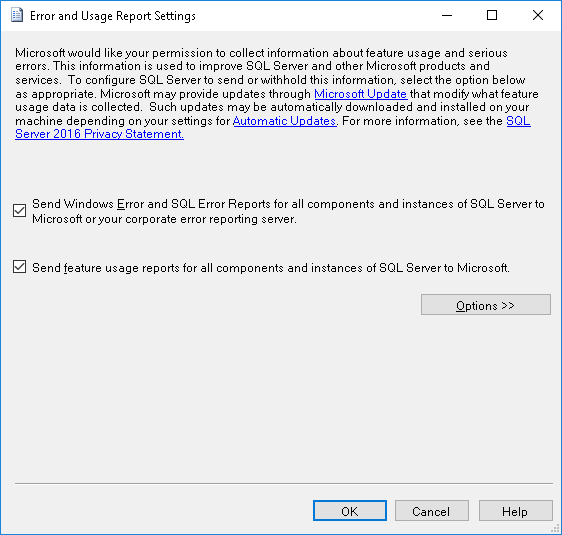
From the Windows Start menu launch SQL Server 2016 Error and Usage Reporting



If prompted by User Account Control “Do you want allow this app to make changes to the device?” (as illustrated below), click **Yes**



The Error and Usage Report Settings dialog is displayed.



Uncheck Send Windows Error and SQL Error Reports for all components and instances of SQL Server to Microsoft or your corporate error reporting server.

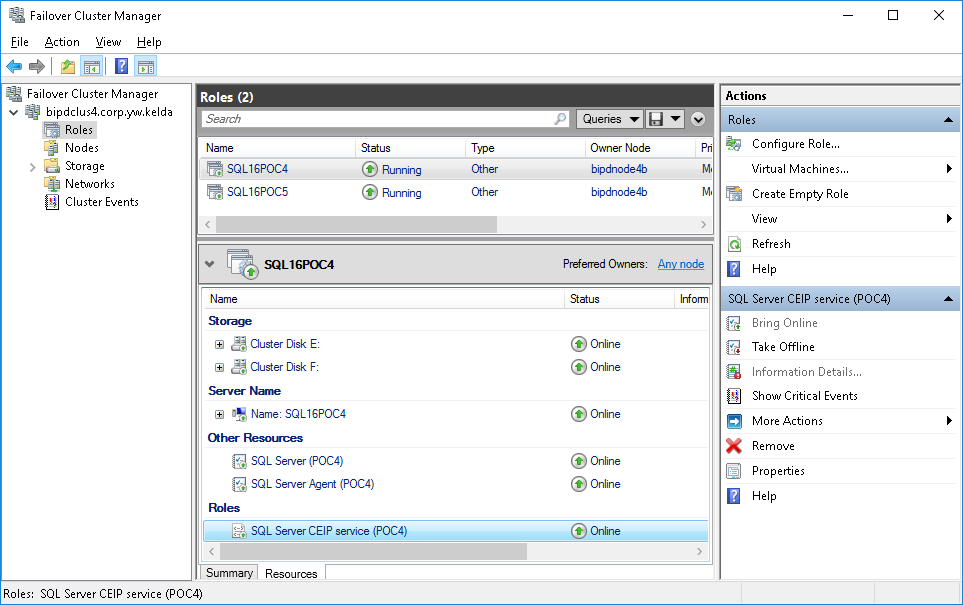
Uncheck Send feature usage reports for all components and instances of SQL Server to Microsoft.

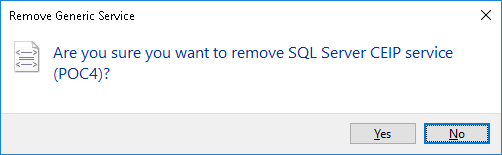
Click **OK**

### Remove the SQL Server CEIP Role

Verify that the steps described in “Configure SQL Server Error and Usage Reporting” above have been completed on *all* cluster nodes before proceeding.

* Start Failover Cluster Manager.
* Navigate to Roles in the left-hand tree view.
* Select a SQL Server role (resource group) in the upper half of the central panel.



* In the lower half of the central panel select the Resources tab.
* Locate the CEIP role, this will be named *SQL Server CEIP (<Instance ID>)* where <Instance ID> is the instance name or MSSQLSERVER for a default instance.
* Right click the *SQL Server CEIP* and select Remove from the pop-up menu.
* A confirmation prompt will be displayed.
* 
* Click Yes
* Repeat for each SQL Server role (resource group) in a multi-instance failover cluster.

Alternatively, start PowerShell as an Administrator and execute:

Remove-ClusterResource -Name "SQL Server CEIP (<Instance ID>)"

Replacing Instance ID with the id of an instance within the cluster, for example

Remove-ClusterResource -Name "SQL Server CEIP (POC5)"

A confirmation message will be displayed:

Remove-ClusterResource

Are you sure you want to remove cluster resource 'SQL Server CEIP (POC5)'? The resource will be taken offline.

[Y] Yes [N] No [S] Suspend [?] Help (default is "Y"):

Type Y and hit return.

Once the clustered CEIP services have been removed from the cluster configuration the underlying Windows services need to be stopped and disabled on *each* node of the cluster.

Log on to each node of the cluster in turn with administrative rights.

Stop the following Windows Services (if present) and change their Startup type to disabled:

* **SQL Server CEIP service (<Instance ID>)** where <Instance ID> is the instance name or MSSQLSERVER for a default instance. In a multi-instance cluster ensure this is repeated for all installed instances.
* **SQL Server Integration Services CEIP service 13.0**

Ensure that this is repeated

Alternatively, start PowerShell from an administrative command prompt and execute the following for a named SQL Server instance:

Set-Service 'SQLTELEMETRY$<Instance ID>' -StartupType 'Disabled'

Set-Service 'SQLTELEMETRY$<Instance ID>' -Status Stopped

Replacing <Instance ID> with the id of the installed SQL Server instance, for example:

Set-Service 'SQLTELEMETRY$POC4' -StartupType Disabled

Set-Service 'SQLTELEMETRY$POC4' -Status Stopped

Or for a default instance of SQL Server:

Set-Service 'SQLTELEMETRY' -StartupType 'Disabled'

Set-Service 'SQLTELEMETRY' -Status Stopped

Repeat these commands for each installed instance of SQL Server.

If SQL Server Integration Services component *SQL Server Integration Services CEIP service 13.0* has been installed execute the following command on each node of the cluster:

Set-Service 'SSISTELEMETRY130' -StartupType Disabled

Set-Service 'SSISTELEMETRY130' -Status Stopped

## Configure SQL Server Integration Services --- To Do

This section should be followed is the Integration Services shared feature has been installed; if it has not this section can be skipped.

Some tasks in this section require **SQL Server Management Studio** (SSMS), either complete the installation of SSMS by referring to Install SQL Server Management Studio below or using a remote connection from your Administrative PC*. The version of SSMS used but be version 16 or later.*

If this document has been followed the *SQL Server Integration Services 13.0* will, in most cases, be configured with a Startup Type of Disabled *unless* a 3rd party requires the service to be running. Verify that this is the case via the Services Control Panel on each cluster node.

If the *SQL Server Integration Services 13.0* service is not required to be running andhas accidentally been installed with a Startup Type of either Manual or Automatic, stop and disable the service via the Control Panel or use the PowerShell below, this will need to be done on each node of the failover cluster.

# Commands to stop and disable the

# SQL Server Integration Services 13.0 service

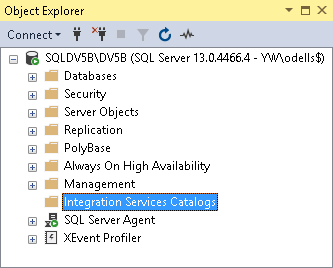
Set-Service 'MsDtsServer130' -Status Stopped

Set-Service 'MsDtsServer130' -StartupType Disabled

Once the Integration Services service configuration is completed the *Integration Services Catalog* can be created.

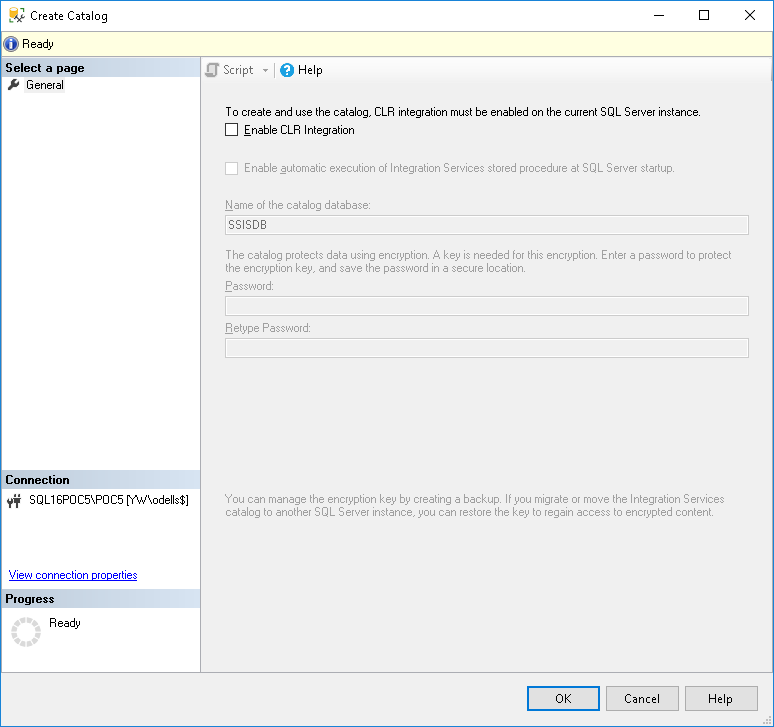
Connect to the instance using SQL Server Management Studio 16 or later.

In the **Object Explorer** select the **Integration Services Catalogs** folder.



Right-click on the Integration Services Catalogs folder and select Create Catalog…

The **Create Catalog** wizard is displayed.



Check the **Enable CLR Integration** checkbox.

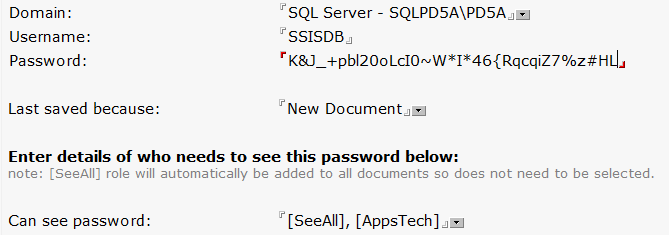
Additional controls will now be enabled. Check the **Enable automatic execution of Integration Services stored procedures at SQL Server startup** checkbox.

Generate a complex password and enter this in the **Password** and **Retype Password** edit boxes.

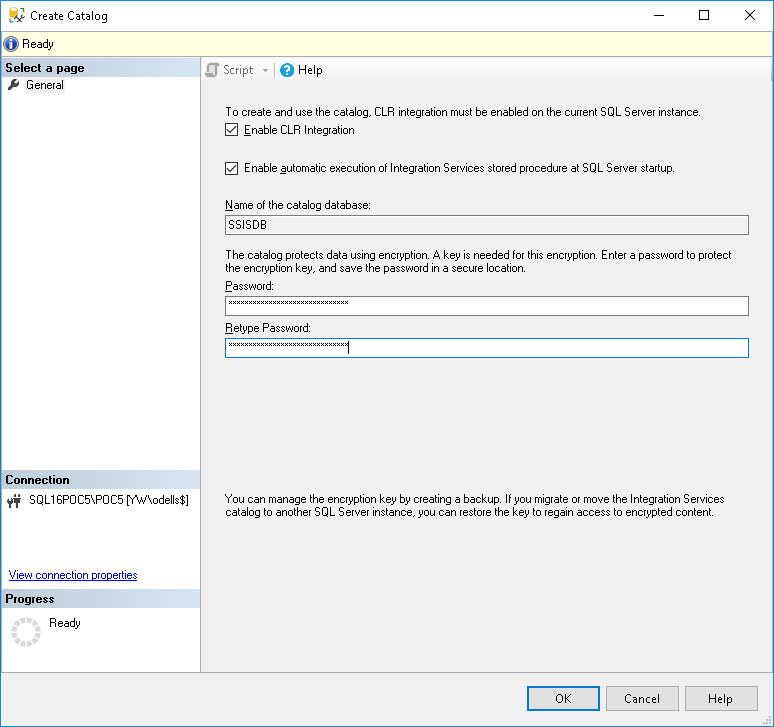
Record this password in the IT Password Repository as:

* Domain: SQL Server - <Instance Name>
* Username: SSISDB
* Can see password: SeeAll, AppsTech

An example from the IT Password Repository is shown below:



Once the password has been safely recorded, return to the **Create Catalog** wizard.



Click **OK**

The **Create Catalog** wizard will show progress for a few seconds and then be dismissed. The Cataog has now been created.

The **SSISDB** catalog is shown under **Integration Services Catalogs** and as a user database.

If the instance is not going to use the *YW SQL Server Default Build* the following SQL statements must be executed now, if the *YW SQL Server Default Build* is going to be installed these commands are included in that build and you may proceed to the Additional Tasks.

If the *YW SQL Server Default Build* is not being installed create a new query in SQL Server Management Studio connected to the instance being configured and execute the following commands:

USE [master]

GO

ALTER DATABASE [SSISDB] SET RECOVERY SIMPLE WITH NO\_WAIT;

ALTER DATABASE [SSISDB] MODIFY FILE ( NAME = N'data', SIZE = 512MB , FILEGROWTH = 0);

ALTER DATABASE [SSISDB] MODIFY FILE ( NAME = N'log', SIZE = 128MB , MAXSIZE = UNLIMITED, FILEGROWTH = 0);

ALTER AUTHORIZATION ON DATABASE::[SSISDB] TO [sa];

ALTER DATABASE [SSISDB] MODIFY FILE (NAME=N'data', NEWNAME=N'SSISDB');

ALTER DATABASE [SSISDB] MODIFY FILE (NAME=N'log', NEWNAME=N'SSISDB\_Log');

USE [SSISDB];

GO

EXEC [catalog].[configure\_catalog] @property\_name = N'MAX\_PROJECT\_VERSIONS', @property\_value = 5;

EXEC [catalog].[configure\_catalog] @property\_name = N'RETENTION\_WINDOW', @property\_value = 90;

EXEC [catalog].[configure\_catalog] @property\_name = N'OPERATION\_CLEANUP\_ENABLED', @property\_value = N'TRUE';

EXEC [catalog].[configure\_catalog] @property\_name = N'SERVER\_CUSTOMIZED\_LOGGING\_LEVEL', @property\_value = N'';

EXEC [catalog].[configure\_catalog] @property\_name = N'SERVER\_LOGGING\_LEVEL', @property\_value = 1;

# Additional Tasks

There are further tasks which should now be performed, these are documented separately.

## Service Packs and Cumulative Updates

In most cases, there will be a Service Pack (SP) and Cumulative Update (CU) to install to update the SQL Server installation; ideally the most recent SP and CU for that release branch should be chosen but this may not be the case if a third-party vendor does not certify their application against a recent update.

For details on SP and CU installation see “How To: Install SQL Server 2016 – Service Pack & Cumulative Updates”

## SQL Server Management Studio

SQL Server Management Studio is no longer shipped with the main SQL Server Setup but as a separate download. If a local installation of SQL Server Management Studio is required on the server refer to “*How To: Install SQL Server Management Studio*”

If installing SQL Server Management Studio it must be installed on each node of the cluster.

## Install TSM & TDP

All physical host servers will require an installation of the TSM Backup Archive client to perform Operating System and file-level backups.

Best practice also requires a TSM Backup Archive client to be installed on virtual machines hosting SQL Server for all but the smallest SQL instances.

In addition, database backups are as a matter of preference taken using Tivoli Data Protection (TDP) for SQL Server.

For information on how to install and configure TSM and TDP for SQL Server, see “*How To: Install & Configure TSM & TDP for SQL Server”*.