**Setup AlwaysOn Servers**

The steps to create an Always on cluster are explained below from the following website.

<https://techcommunity.microsoft.com/t5/itops-talk-blog/step-by-step-creating-a-sql-server-always-on-availability-group/ba-p/648772>

Additional steps are required on completion.

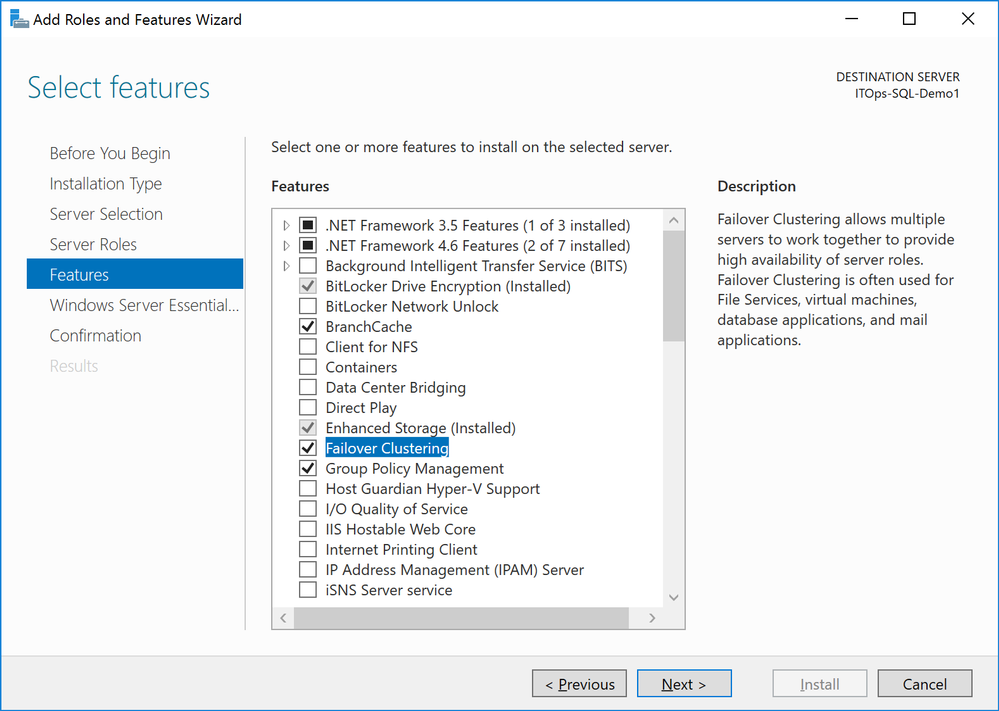
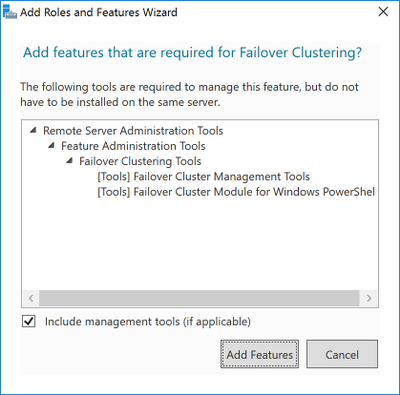
# [Step-By-Step: Creating a SQL Server Always On Availability Group](https://techcommunity.microsoft.com/t5/itops-talk-blog/step-by-step-creating-a-sql-server-always-on-availability-group/ba-p/648772)

One of the most customer requested features in SQL, from as far back as [SQL 2012](https://blogs.technet.microsoft.com/canitpro/2013/08/19/step-by-step-creating-a-sql-server-2012-alwayson-availability-group/?WT.mc_id=ITOPSTALK-blog-abartolo), has been [Always On Availability Groups](https://docs.microsoft.com/en-us/sql/database-engine/availability-groups/windows/overview-of-always-on-availability-groups-sql-server?view=sql-server-2017&viewFallbackFrom=sql-server-2017%3FWT.mc_id%3DITOpsTalk-blog-abartolo). This takes disaster recovery and high availability to a new level by enabling multiple copies of the database to be highly available, enabling the possibility of Read-Only workloads and enabling the ability of offloading management tasks such as backups.  Unlike database mirroring, Always On Availability Groups allow for failover of a group of databases in a single SQL Server instance.

This step-by-step details the creation of a SQL Server 2016 / 2017 Always On Availability Group to ensure high availability of mission-critical databases.

## **Step 1: Windows Failover Cluster Feature Installation**

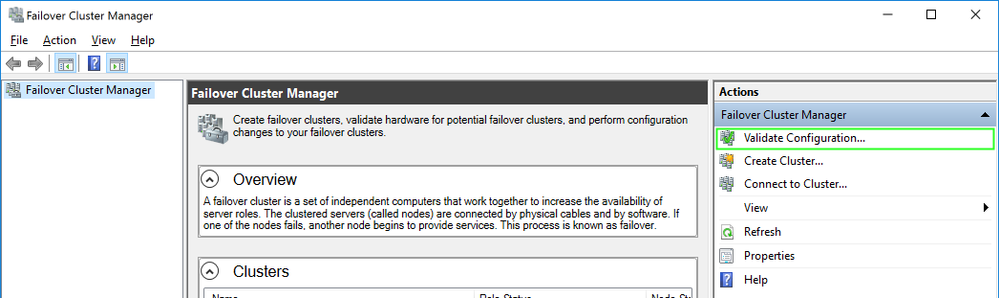
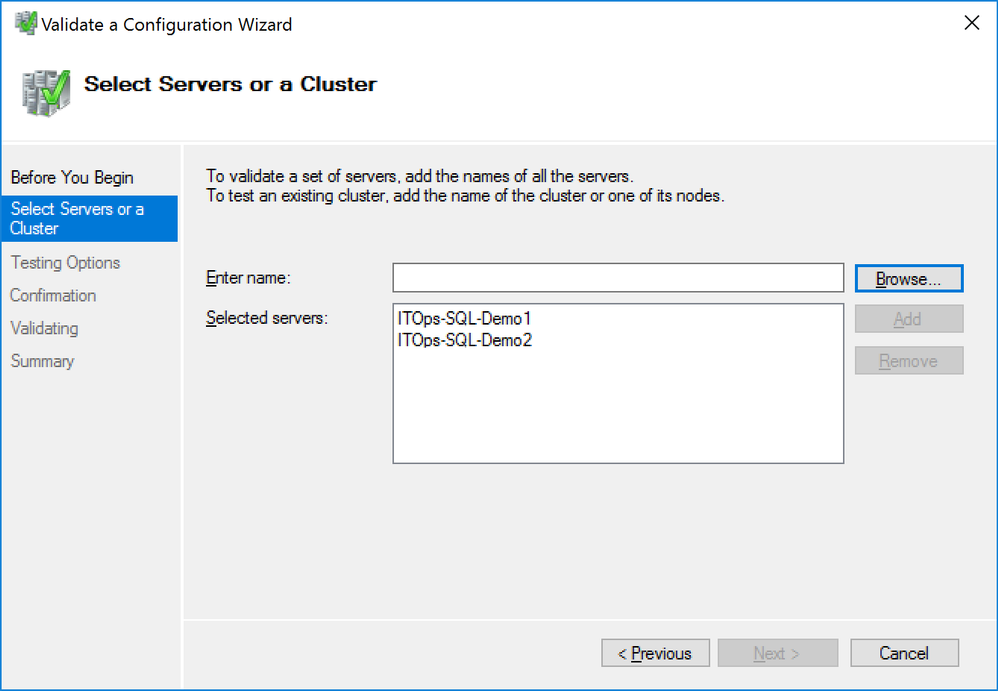
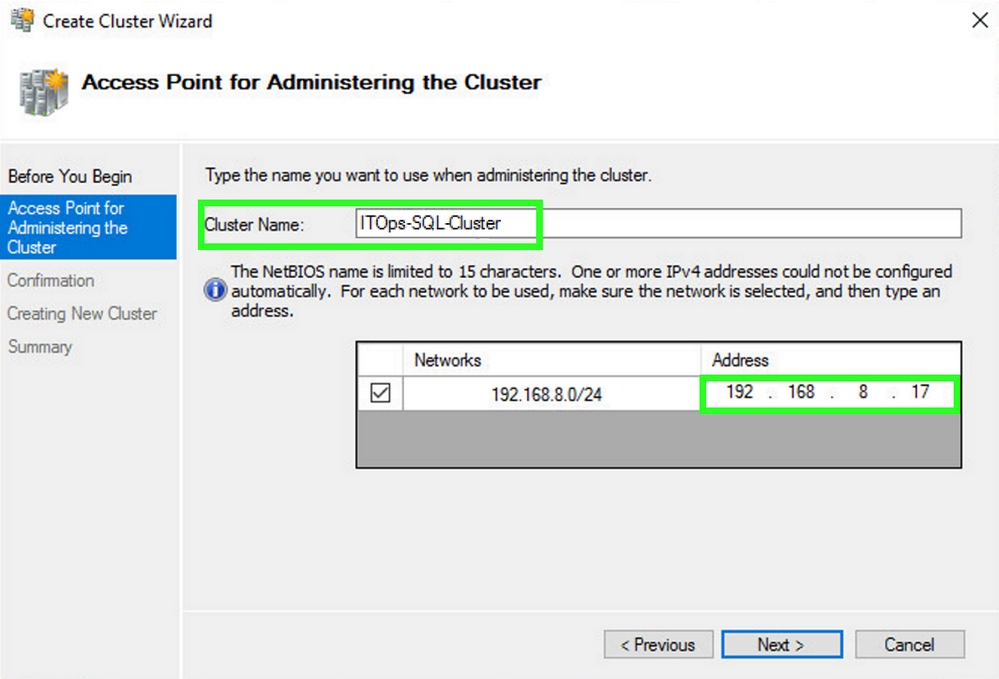
The Windows Failover Cluster feature needs to be added to all machines running SQL Server instances that will be configured as replicas.  The following steps will add the feature to machines running Windows Server 2016:

1. Run the Server Manager console and select Add roles and features.
2. Once the Add Roles Features Wizard has launched, click Next until the Select Features dialog box has been reached.
3. Select the Failover Clustering checkbox.  
      
   *Add Roles and Features Wizard*
4. Select Add Features within the Add features that are required for Failover Clustering dialog box and click Next.  
      
   *Add Roles and Features Wizard*
5. Click Next until the Confirm installation selections dialog box is reached and select Install.

## **Step 2: Enabling Windows Failover Clustering Configuration for SQL Server**

**At this point, proper Active Directory rights are required to configure the Windows Server Failover Cluster. Microsoft provides a complete listing of different Active Directory permissions able to create a Windows Server Failover Cluster. Take a moment to review the**[Step-By-Step Guide to Configuring Accounts in Active Directory](https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2008-R2-and-2008/cc731002(v=ws.10)?WT.mc_id=ITOpsTalk-blog-abartolo)**for more details.**

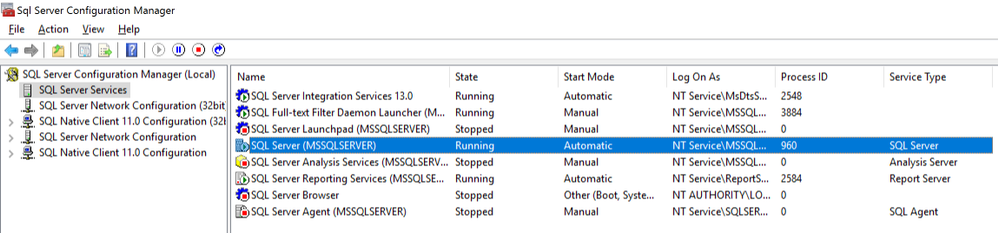
**The following steps will configure Windows Failover Clustering:**

1. Run Failover Cluster Manager found under Tools (top right) within the Server Manager Console.
2. Click Validate Configuration inside the Actions box.  
      
   *Failover Cluster Manager*
3. Click Next in the Validate a Configuration Wizard: Before You Begin dialog box.
4. Add the server hostnames of the SQL Server instances that you want to configure as replicas in the Availability Group and click Next.  
      
   *Validate a Configuration Wizard*
5. Ensure the Run all tests option is selected in the Testing Options dialog box and click Next.
6. Click Next in the Confirmation dialog box.
7. Select Finish in the Summary dialog box to create the Windows Failover Cluster.  
      
   NOTE: It is expected that the Failover Cluster Validation Wizard will report several warnings especially if shared storage is not utilized. Be aware that these warning messages will configure a file share witness for said cluster quorum configuration. An error messages reported will need to be addresses prior to the creation of the Windows Server Failover Cluster.
8. Enter the server name and IP address of said Windows Server Failover Cluster in the Access Point for Administering the Cluster dialog box.  
      
   *Create Cluster Wizard: Access Point for Adminstrating the Cluster*
9. Click Next in the Confirmation dialog box to create the Windows Failover Cluster using the servers as nodes of the cluster. Be sure to include the required DNS and Active Directory entries for the cluster hostname.
10. Verify that the configuration is successful within the Summery dialog box.
11. Select More Actions > Configure Cluster Quorum Settings... to configure the cluster quorum configuration to use a file share.  The wizard will configure the cluster to use Node Majority by default.
12. Click Next.
13. Select Add or change the quorum witness option in the Select Quorum Configuration page and click Next.
14. Select Configure a file share witness option in the Select Quorum Witness page and Next.
15. Type the path of the file share needed to use in the File Share Path: text box within the Configure File Share Witness page and click Next.
16. Click Next in the Confirmation page.
17. Click Finish in the Summary page.

## **Step 3: Enabling the SQL Server 2016 Always On Availability Groups Feature**

Work can now start in enabling the Always On Availability Groups feature in SQL Server 2016 now that the Windows Server Failover Cluster has been created. All SQL Server instances need to be configured as replicas in said Availability Group.

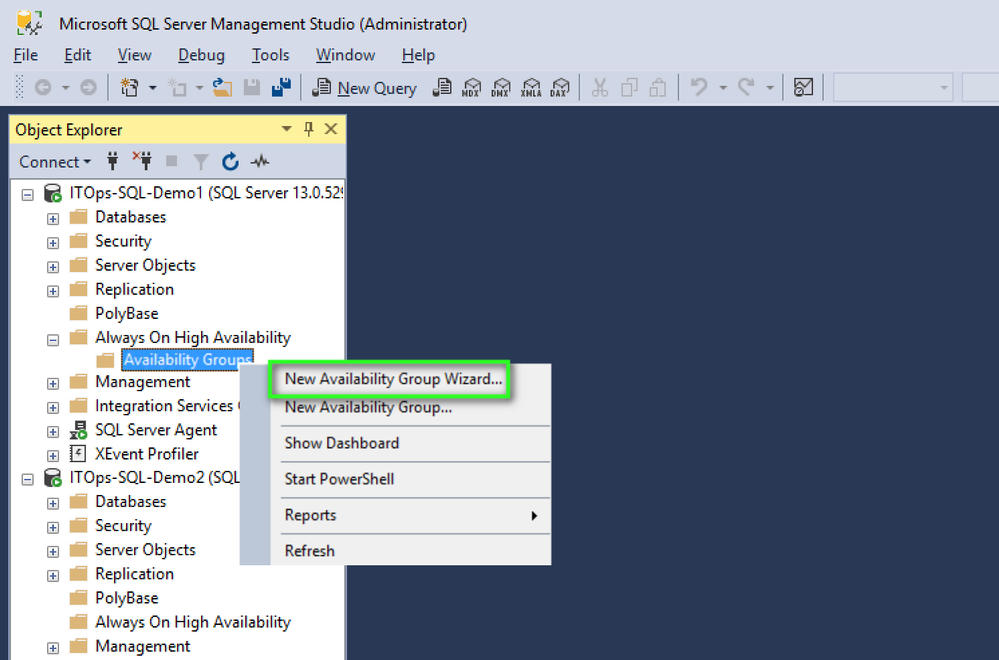
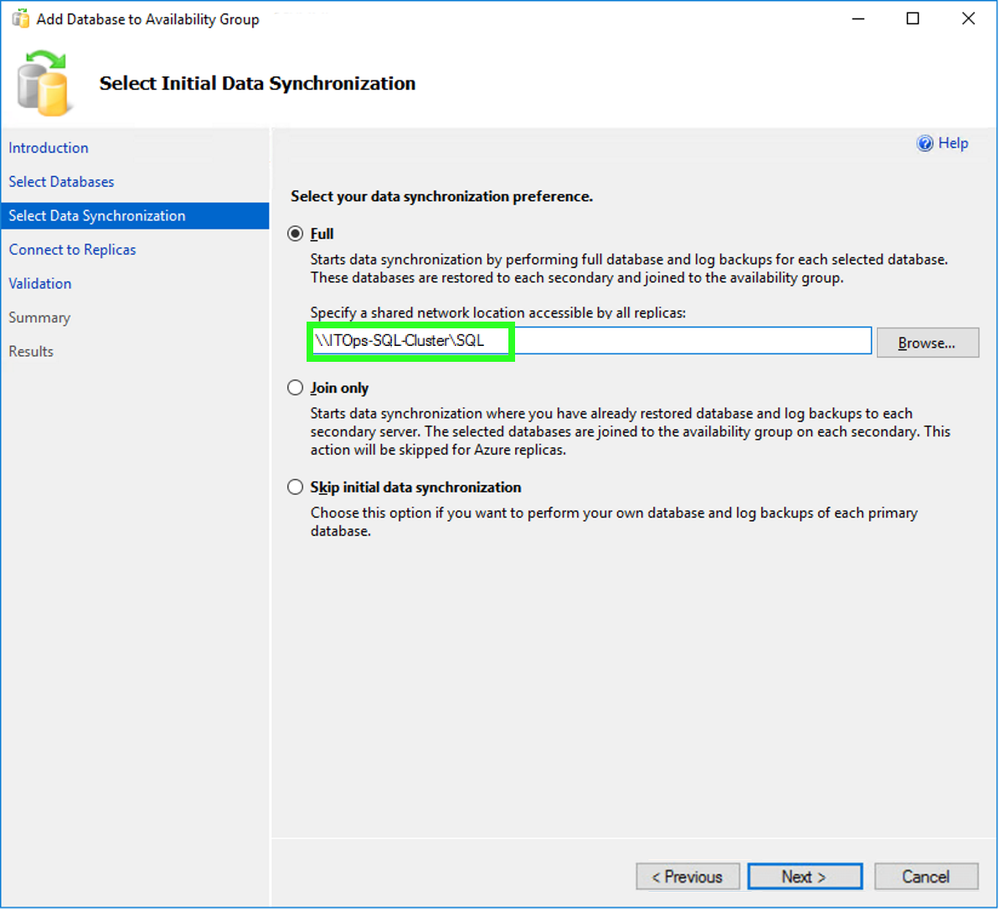
**The following steps will enable the Always On Availability Group feature:**

1. **Run the SQL Server Configuration Manager and double-click the SQLServer (MSSQLSERVER) service to open the Properties dialog box.**  
     
   *SQL Server Configuration Manager*
2. **Select the Always On High Availability tab in the Properties dialog box and check off the Enable Always On Availability Groups checkbox.**
3. Click OK when prompted to restart the Server service.

**Step 4: Creating and Configuring SQL Server Always On Availability Groups**

Availability Groups can be created on temporary databases or on existing ones. Some solutions, like SharePoint, require the creation of a temporary database so that the solution can harness the AlwaysOn Availability Group when creating the admin content databases and farm configuration.  The temporary database is then removed from the Availability Group configuration once the SharePoint farm is created.

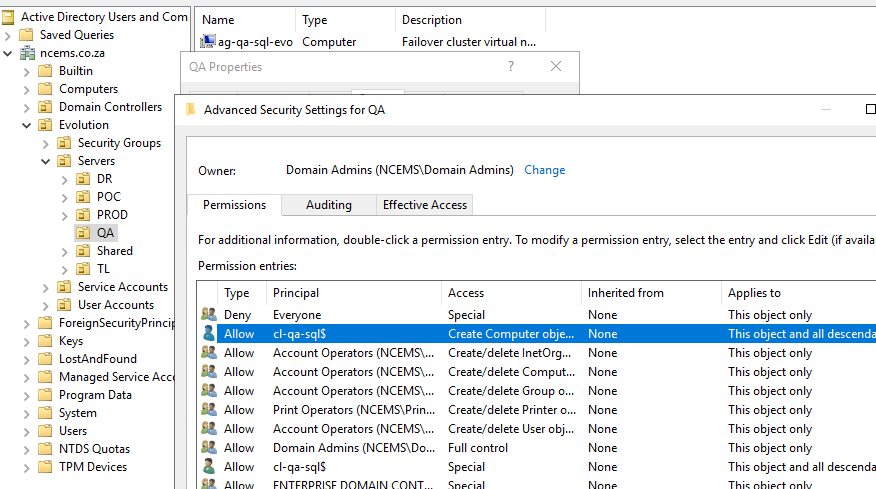
**The following steps will create and configure a SQL Server Always On Availability Group:**

1. **Open SQL Server Management Studio and connect to the SQL Server instance.**
2. **Expand the Always On High Availability folder in the Object Explorer.**
3. **Right-click on the Availability Groups folder and select the New Availability Group Wizard... option launching the New Availability Group Wizard.**  
     
   *SQL Server Management Studio*
4. **Click Next on the Introduction page.**
5. **Enter the name of the Availability Group in the Availability group name: field and click Next.**
6. **Select the checkbox beside the database to be included in the Availability Group within the Select Databases page.**  
     
   NOTE:**The selected databases need to be in a Full Recovery model prior to joining them in the Availability Group.**
7. **Click Next.**
8. Under the Replicas tab within the Specify Replicas page, click Add Replicas and connect to the other SQL Server instances previously joined as nodes with the Windows Server Failover Cluster and configure the following options:  
      
   - Automatic Failover (Up to 2):  Checked  
   - Synchronous Commit (Up to 3): Checked  
   - Readable Secondary: No
9. Verify that the port number value is 5022 in the Endpoints tab.
10. Select the Create an availability group listener option within the Listener tab and enter the following details:   
       
    - Listener DNS name: Name that will be used in the application connection string  
    - Port: 1433  
        
    Click Add... once completed and provide the required IP Address.
11. Enter the preferred virtual IP address in the IPv4 Address field within the Add IP Address dialog box and click OK > Next.
12. Select the Full option within the Select Initial Data Synchronization page.   
       
    *New Availability Group: Select Initial Data Synchronization*  
       
    NOTE: This enables a temporary file share to store database backups used to initialize the databases in an Availability group.  The file share folder is accessible to both the SQL service account and the replicas. It is recommended that manual initialization of the databases prior to configuration should be initialized in the case of larger databases as the network may not be able to accommodate the size of the database backups.
13. Click Next.
14. Verify all validation checks are successful in the Validation page and click Next.
15. Verify all configuration settings and click Finish in the Summary page.
16. Verify all task have been completed successfully in the Results page.

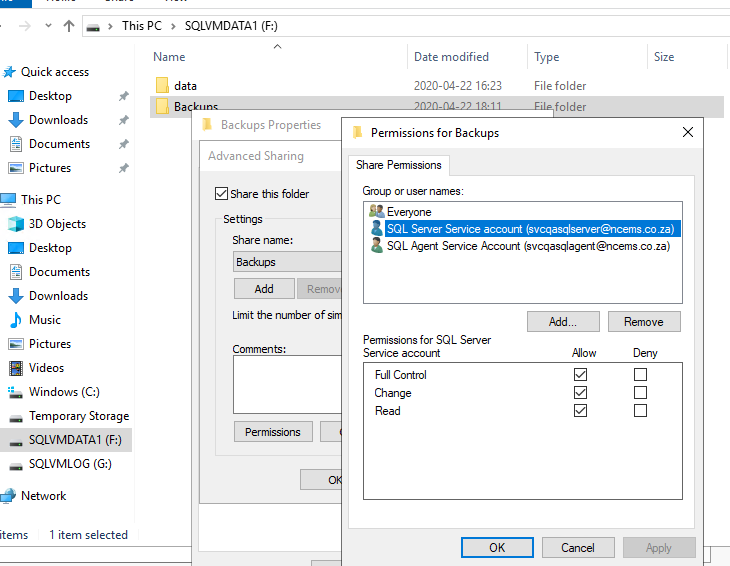
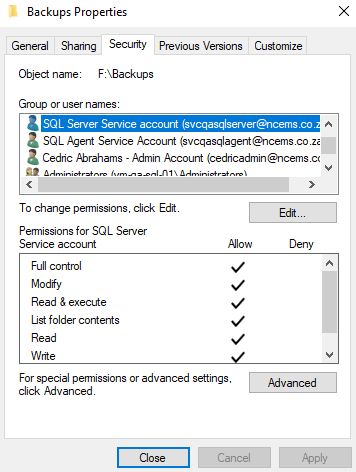
The creation of the SQL Server Always On Availability Group is now complete.  Do note that any additional new databases will need to be manually added.

## **Additional Steps**

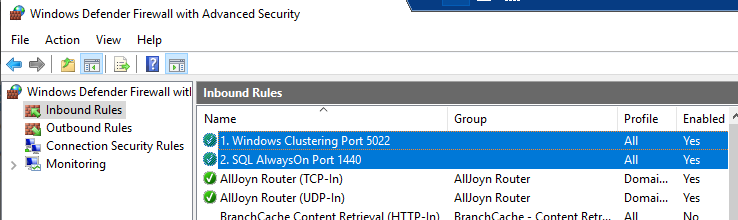
Once the Windows cluster has been created, but before configuring AlwaysOn, you must give the cluster computer account rights to create computer objects in the OU where the cluster servers are.



Create backup folder on SQL-01 and share it giving the SQL service accounts full permission on the folders and the share.

Create Windows Firewall Rues to allow AlwaysOn to communicate. Port 1440 is a custom port assigned during AlwaysOn setup.



netsh advfirewall firewall add rule name= "Windows Clustering Port 5022" dir=in action=allow protocol=TCP localport=5022

netsh advfirewall firewall add rule name= "SQL AlwaysOn Port 1440" dir=in action=allow protocol=TCP localport=1440

Once created you need to run the following to allow for

GRANT ALTER ANY AVAILABILITY GROUP TO [NT AUTHORITY\SYSTEM]  
GO  
GRANT CONNECT SQL TO [NT AUTHORITY\SYSTEM]  
GO  
GRANT VIEW SERVER STATE TO [NT AUTHORITY\SYSTEM]  
GO

**Check Sync Status**

SELECT ag.name AS [AG Name], ar.replica\_server\_name, ar.availability\_mode\_desc, adc.[database\_name],

       drs.is\_local, drs.is\_primary\_replica, drs.synchronization\_state\_desc, drs.is\_commit\_participant,

          drs.synchronization\_health\_desc, drs.recovery\_lsn, drs.truncation\_lsn, drs.last\_sent\_lsn,

          drs.last\_sent\_time, drs.last\_received\_lsn, drs.last\_received\_time, drs.last\_hardened\_lsn,

          drs.last\_hardened\_time, drs.last\_redone\_lsn, drs.last\_redone\_time, drs.log\_send\_queue\_size,

          drs.log\_send\_rate, drs.redo\_queue\_size, drs.redo\_rate, drs.filestream\_send\_rate,

          drs.end\_of\_log\_lsn, drs.last\_commit\_lsn, drs.last\_commit\_time, drs.database\_state\_desc

FROM sys.dm\_hadr\_database\_replica\_states AS drs WITH (NOLOCK)

INNER JOIN sys.availability\_databases\_cluster AS adc WITH (NOLOCK)

ON drs.group\_id = adc.group\_id

AND drs.group\_database\_id = adc.group\_database\_id

INNER JOIN sys.availability\_groups AS ag WITH (NOLOCK)

ON ag.group\_id = drs.group\_id

INNER JOIN sys.availability\_replicas AS ar WITH (NOLOCK)

ON drs.group\_id = ar.group\_id

AND drs.replica\_id = ar.replica\_id

ORDER BY ag.name, ar.replica\_server\_name, adc.[database\_name] OPTION (RECOMPILE);