Technical Report

SQL Server on Google Cloud Platform Using Cloud Volumes Service

Deployment guide

August 2021 | TR-4898
Pat Sinthusan, NetApp

Akaash Rampersad
Customer Engineer, Infrastructure Modernization
Google Cloud

In partnership with

Google Cloud

Abstract

This document provides guidance on how to deploy Microsoft SQL Server leveraging NetApp® Cloud Volumes Service on virtual machines (VMs) in Google Cloud Platform.

TABLE OF CONTENTS

LIST OF FIGURES

Intended audience

This technical report provides an overview of how to configure a highly resilient Google Cloud Platform environment for SQL Server installed on SMB fileshare storage using NetApp Cloud Volumes Service (CVS) for Google Cloud.

This document is intended for NetApp Cloud Volumes Service and/or SQL Server database administrators who are responsible for deploying Microsoft SQL Server in Google Cloud Platform. It is assumed that the reader is familiar with the various components of the solution.

Microsoft SQL Server on Google Cloud Platform overview

Today, many customers migrate to Google Cloud Platform to accelerate their SQL Server deployments, reduce costs, and provide increased agility for their business processes. These key benefits enable IT decision makers to adopt a cloud-first strategy for a critical infrastructure such as SQL Server.

Moreover, moving the SQL Server estate to Google Cloud and running it on Google Compute Engine (GCE) provides full compatibility with existing tools and workflows such as SQL Server Management Studio (SSMS), SQL Server Integration Services (SSIS), SQL Server Reporting Services (SSRS), and Visual Studio.

In preparation for migrating production SQL Servers into the cloud, customers often migrate their development, test, staging, and other nonproduction SQL Servers first to test and certify their systems. GCE offers instances of various sizes along with multiple deployment options to accommodate these various systems. Deployments can scale from small, single host configurations to complex multizone/region configurations.

Memory-optimized GCE instances such as the M1-family and M2-family offer excellent compute capabilities for demanding database workloads such as SQL Server volume location databases (VLDBs) or SAP HANA while other standard workload databases run equally well on general-purpose GCE instances such as the N1-family, N2-family, and E2-family. Block storage for these machine families is provided by Google Cloud Persistent Disks, which are networked together and provide IOPs based on the size of the disk along with the number of vCPUs on the instance. NetApp Cloud Volumes can help overcome these conditions by providing IOPs at different performance levels regardless of the size of the volume.

Traditional SQL Server Always On instances usually require each node to maintain a copy of the data, which can quickly become costly as the databases grow. For example, in a traditional SQL Server Always On deployment, if the databases are 10TB in size, each replica must have at least 10TB of storage provisioned (primary replica, 10TB; secondary replica, 10TB; and so on). One of the benefits of having a Windows Server Failover Cluster (WSFC) deployed for SQL Server is only needing to have one copy of the data, which is shared between the nodes of the cluster. Therefore, running SQL Server Always On in a failover cluster instance (FCI) on a WSFC can immediately realize cost savings by reducing the amount of storage required.

NetApp Cloud Volumes Service and Cloud Volumes ONTAP are two cloud storage solutions for running high performance SQL Server workloads in combination with GCE instances.

Before you begin

Before you complete the steps in this report, you should already have:

- A Google Cloud Platform subscription
- A Windows Active Directory domain

- A domain user account to run the SQL Server service and login capability into the GCE instance to mount file shares
- A domain user account for installing SQL Server
- DNS configured on the Google Cloud Platform network, pointing to the domain controller

Prerequisites

You should have a domain user account that has permissions to create objects on both GCE instances and in Active Directory.

You should also have an account to create Cloud Volumes Service and a basic understanding of how to install SQL Server single instance.

Before you create a volume in Cloud Volumes Service, you should a solid understanding of Cloud Volumes Service service levels and CVS-Performance service type. For information about service-level performance of CVS-Performance volumes, see the following resources:

- Performance expectations
- Selecting the appropriate service level and allocated capacity for NetApp Cloud Volumes Service

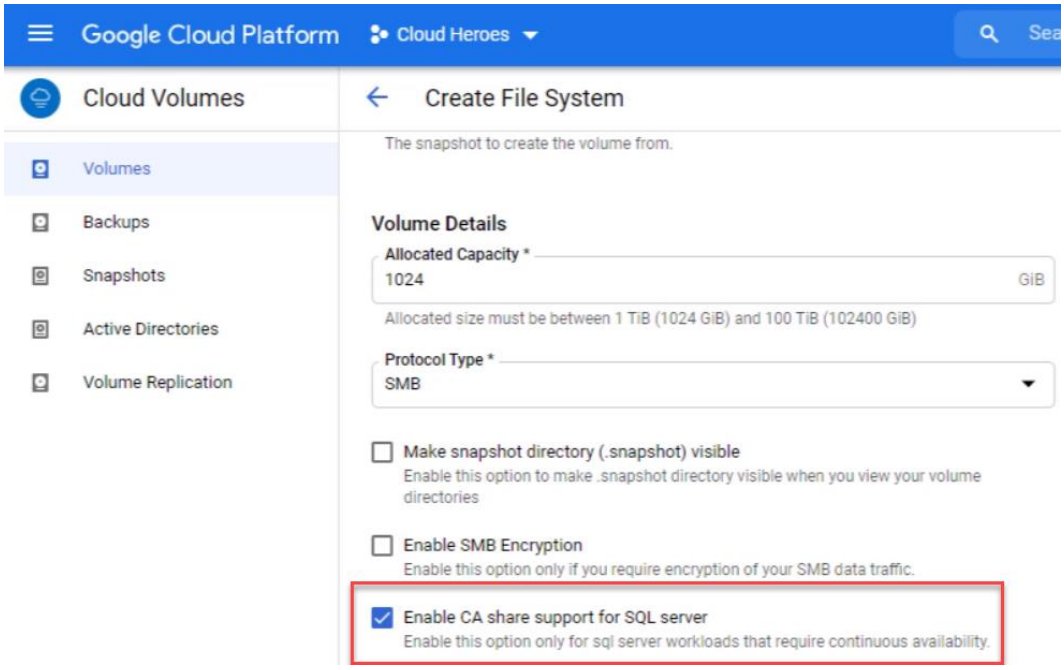
SMB and continuous availability

SMB3 with the continuously available share property enabled provides a very high level of resiliency between the GCE instances and the storage service. SMB Transparent Failover enables maintenance operations on the Cloud Volumes Service without interrupting connectivity to server applications storing and accessing data on SMB volumes. To support SMB Transparent Failover, Cloud Volumes Service supports the SMB continuous availability (CA) shares option for use with SQL Server applications over SMB running on GCE instances. SMB CA shares enable SQL Server workloads on Cloud Volumes Service, which provides performance improvements, scale, and cost benefits for single instance, Always On failover cluster (AOFC) instance, and Always On availability group deployments.

How to enable continuous availability on Cloud Volumes Service

NetApp recommends that you enable CA for all the volumes that host the SQL Server data files. This option can be set during volume creation, as shown in Figure 1.

Figure 1) Enable CA for all volumes.



SQL Server over SMB requirements

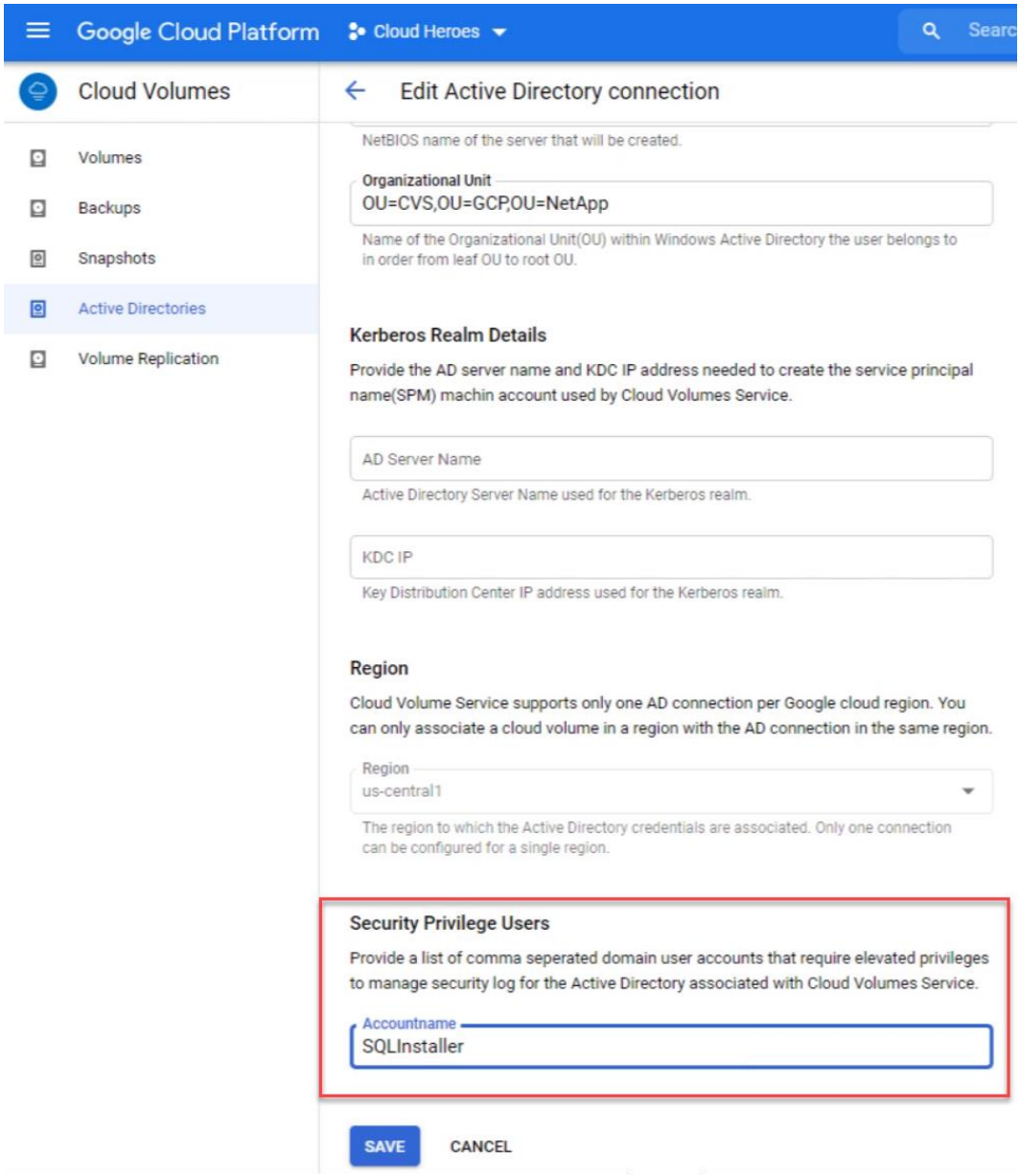
Starting with SQL Server 2012 (11.x), system databases (master, model, MSDB, and TempDB) and database engine user databases can be installed with Server Message Block (SMB) file server as a storage option. This applies to both SQL Server stand-alone and SQL Server FCIs.

These allow you to leverage Cloud Volumes Service with all its performance and data management capabilities, such as volume capacity, performance scalability, and data protection features, which the SQL Server can take advantage of.

To install SQL Server over SMB shares, the installer must meet the following requirements:

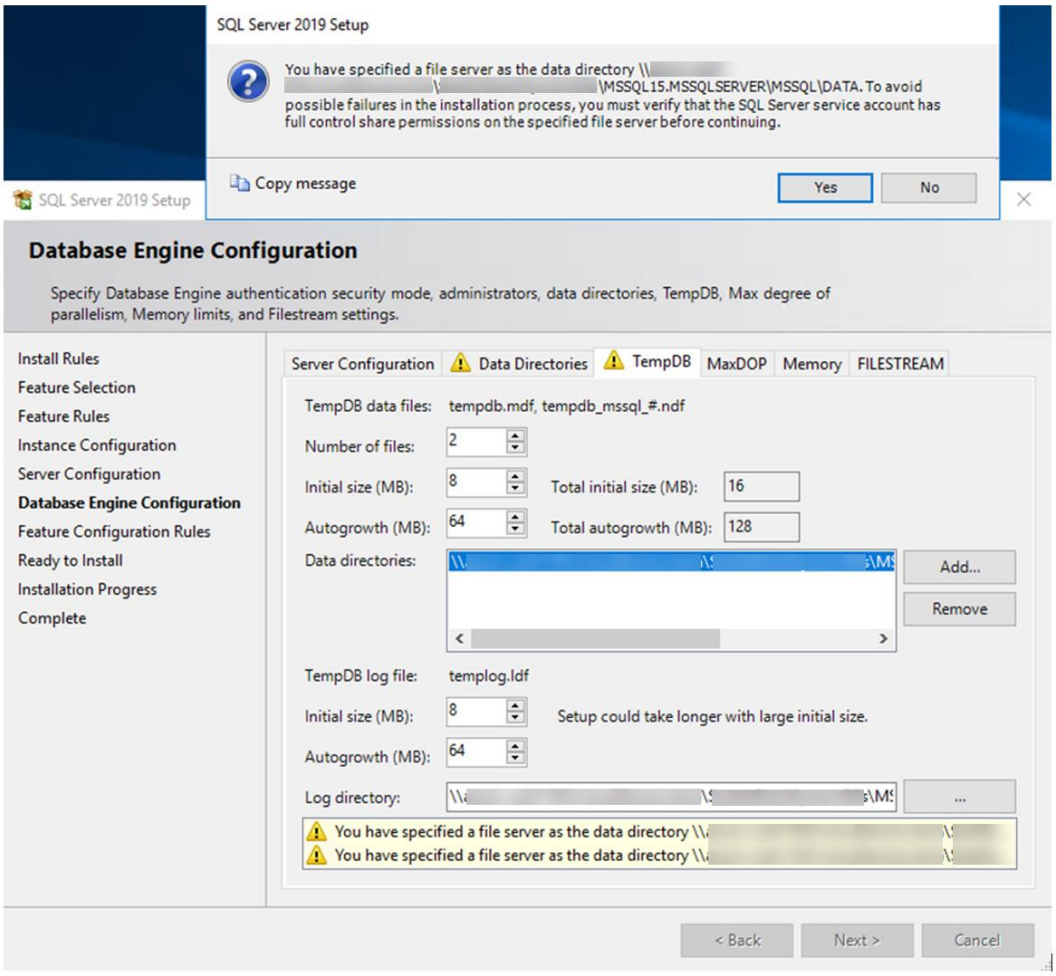
- The installer has been granted the SeSecurityPrivilege setting through Google Cloud Platform Portal.
- The installer has read/write access to the share.
- The installer is a member of the local Administrators Group on the Windows host.

Figure 2) Join Active Directory and provide domain account which elevated privilege.



The installation process is the same as for the typical block storage. The only exception is that the data root directory for system database files can be pointed to an SMB share during the database engine configuration step.

Figure 3) Specify CVS path for system database files to reside.

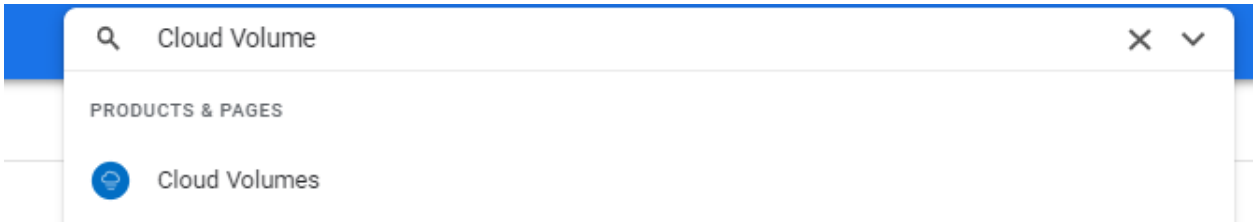


Create Cloud Volumes Service for SQL Server

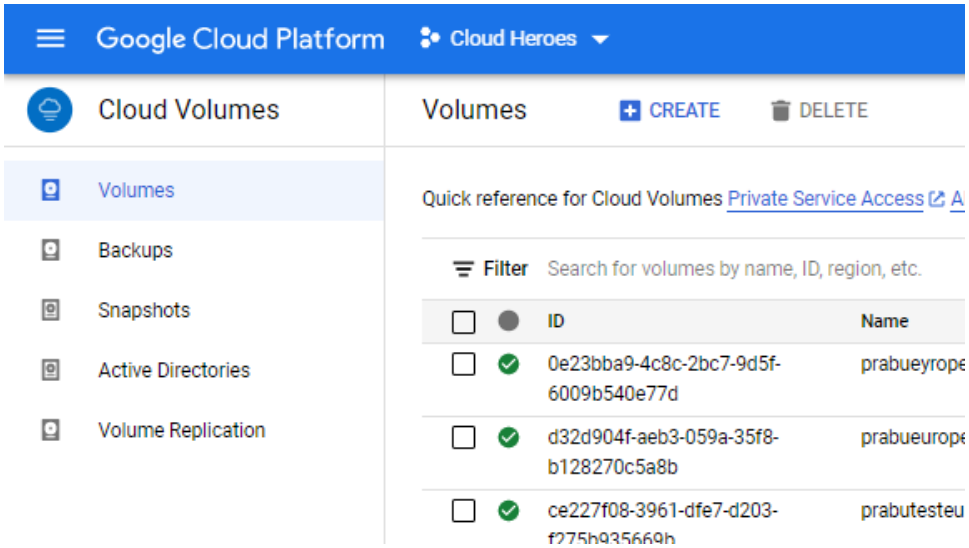
Create a Cloud Volumes Service volume

To create a Cloud Volumes Service volume, complete the following steps:

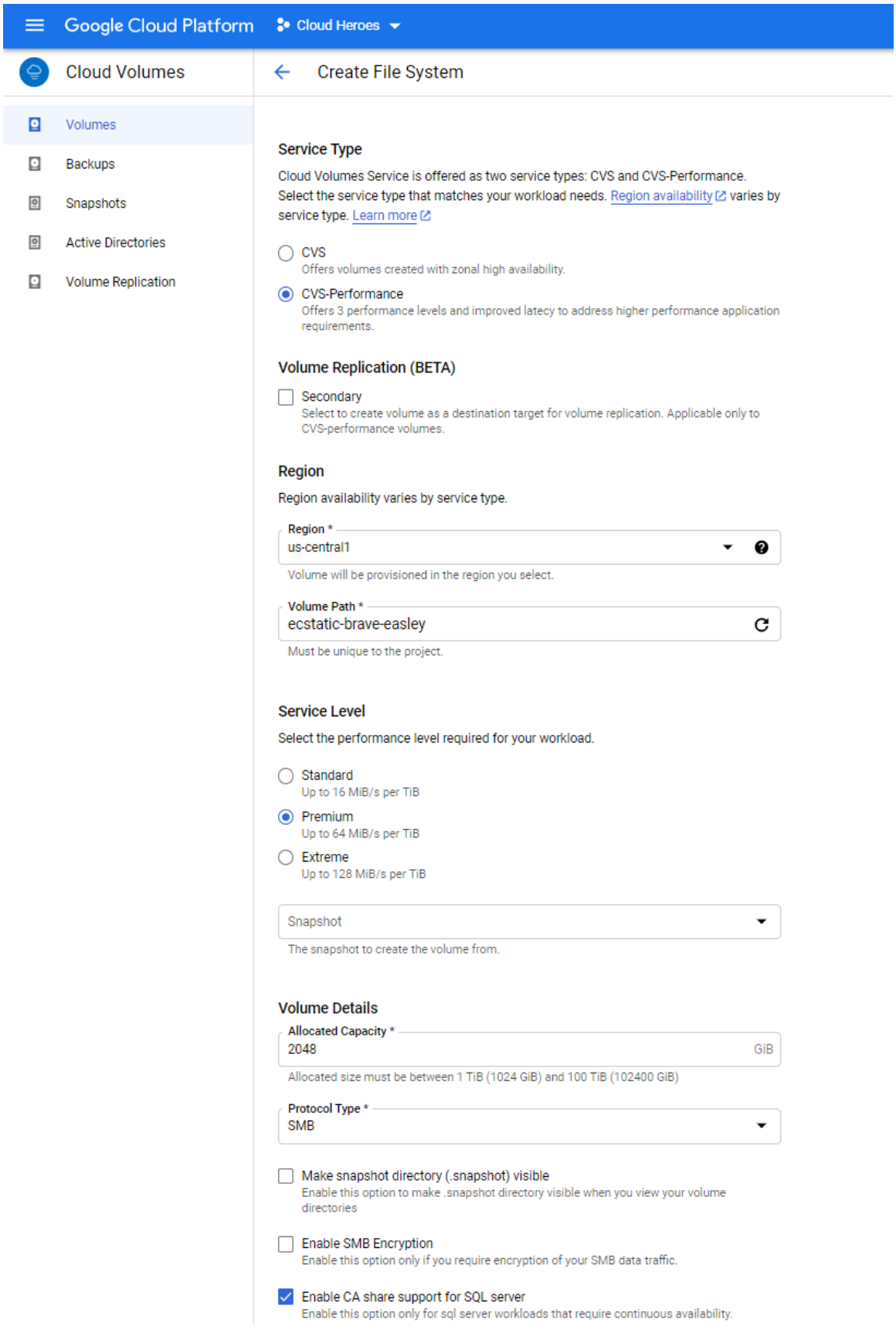
1. From the Google Cloud Platform portal, select Cloud Volumes Service or enter Cloud Volumes Service in the search box.



2. To create an SMB volume, select Volumes and Create.



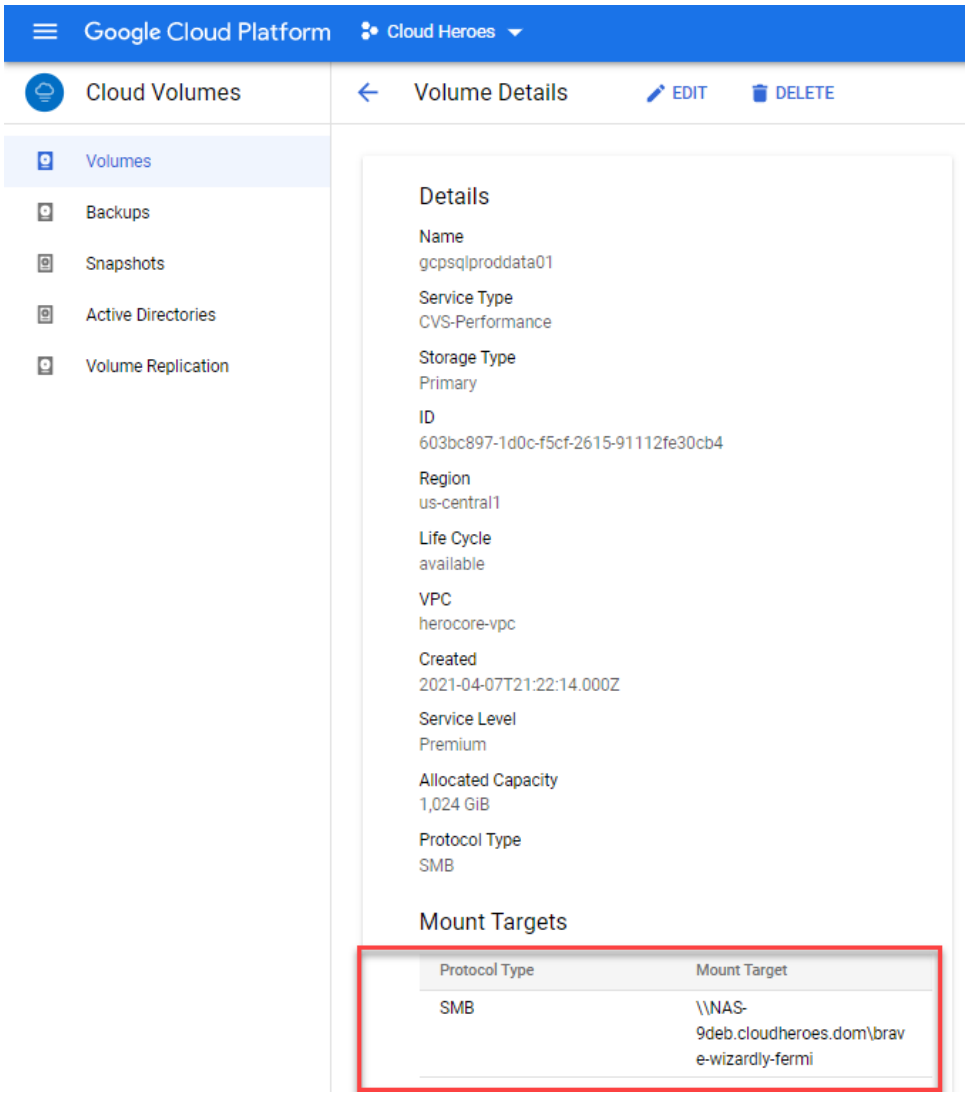
3. Provide the volume name, service type, region, service level, allocated capacity, and protocol type. Enable CA share support for SQL Server, then click Save.



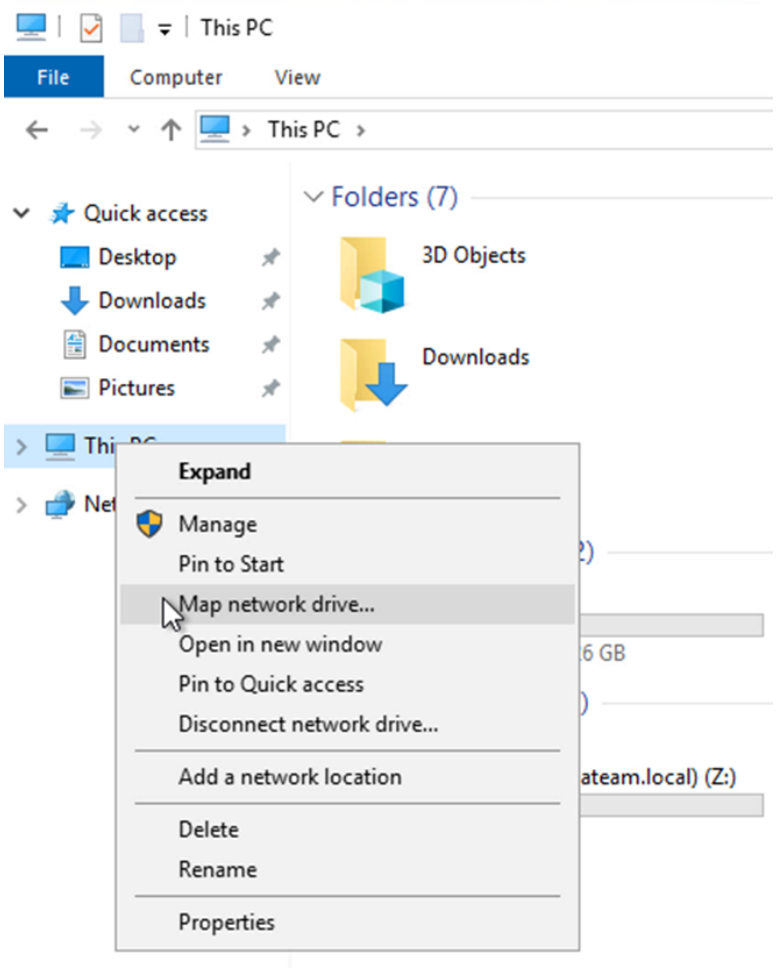
Assign permissions and create a database over SMB on Cloud Volumes Service

To assign permissions and create a database over SMB on Cloud Volumes Service, complete the following steps:

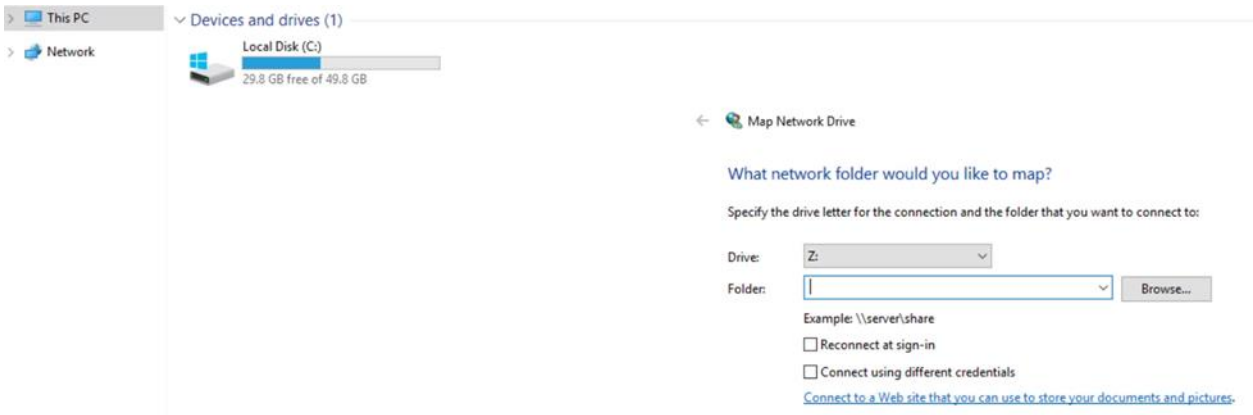
1. From the Google Cloud console, select Cloud Volumes and select the volume that was just created.



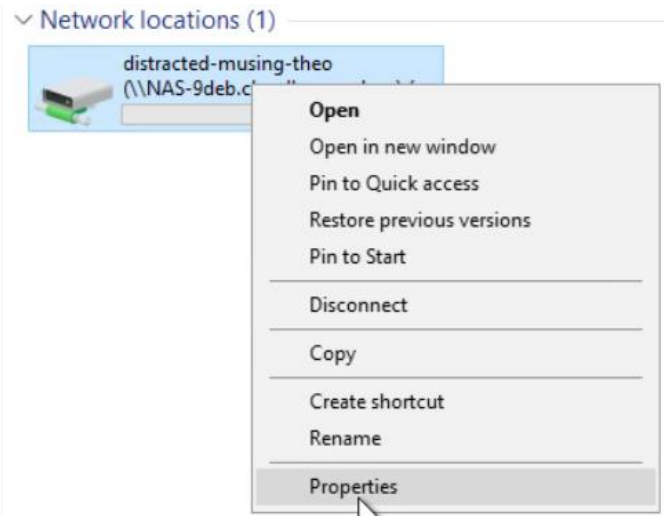
2. On the Google Cloud Platform VM, open Windows Explorer.
3. Right-click This PC.
4. Select Map Network Drive.



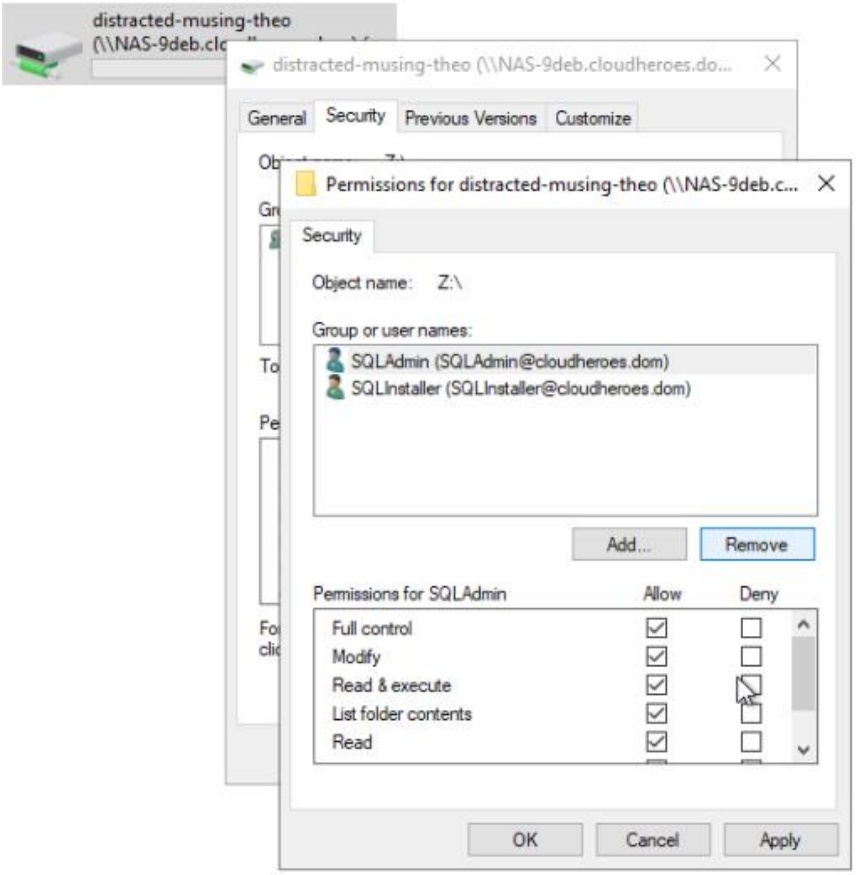
5. Paste the copied SMB path in the folder.



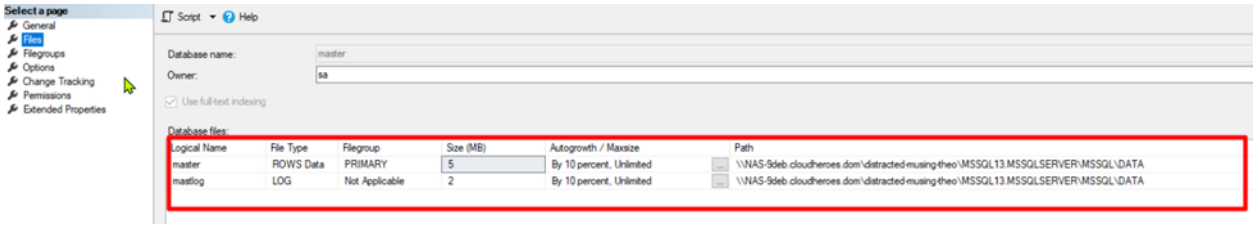
6. At sign-in, disable the Reconnect option.
7. Click Finish.
8. Right-click the mapped network drive and select Properties.



9. From the Security tab, click Edit.
10. Remove Everyone and add the SQL Server installer and SQL Server service accounts with full control permissions.



11. Create the database with data and log files residing on the Cloud Volumes Service SMB shares by using SQL Server Management Studio (SSMS).



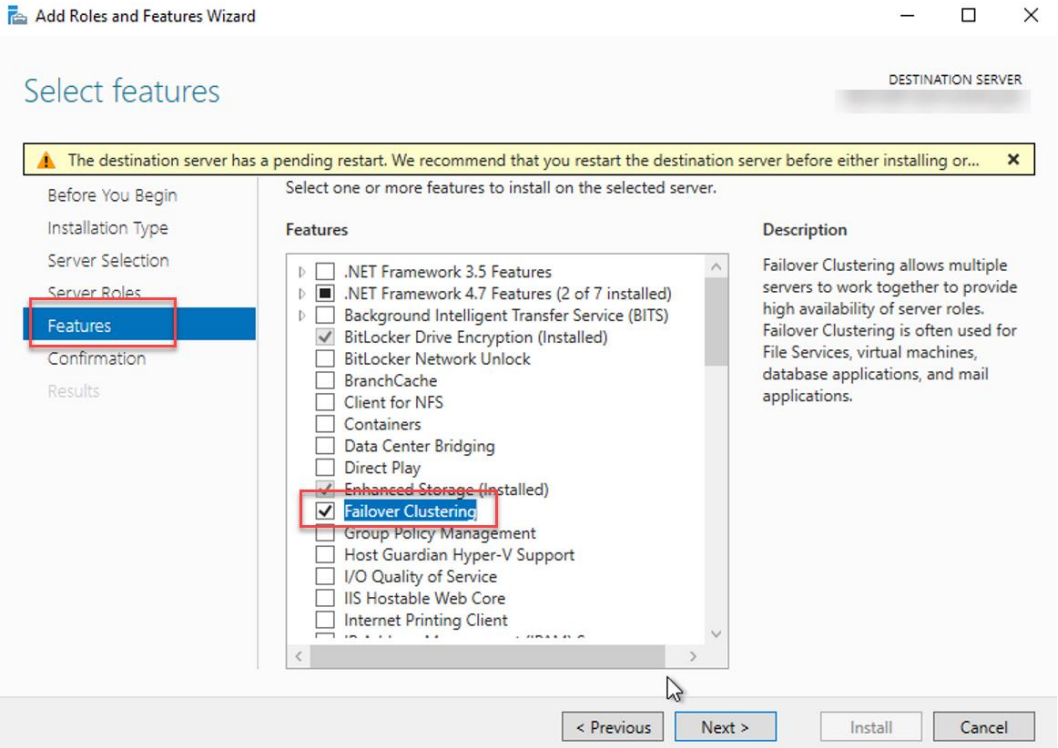
Deploy Always On failover cluster over SMB on Cloud Volumes Service

Google Cloud and NetApp recommend creating GCE instances in different zones. The following section describes how to deploy AOFC over SMB on Cloud Volumes Service.

Create Windows Server Failover Cluster

To create WSFC, complete the following steps:

1. Using Server Manager, add the Failover Clustering feature. Click Next.



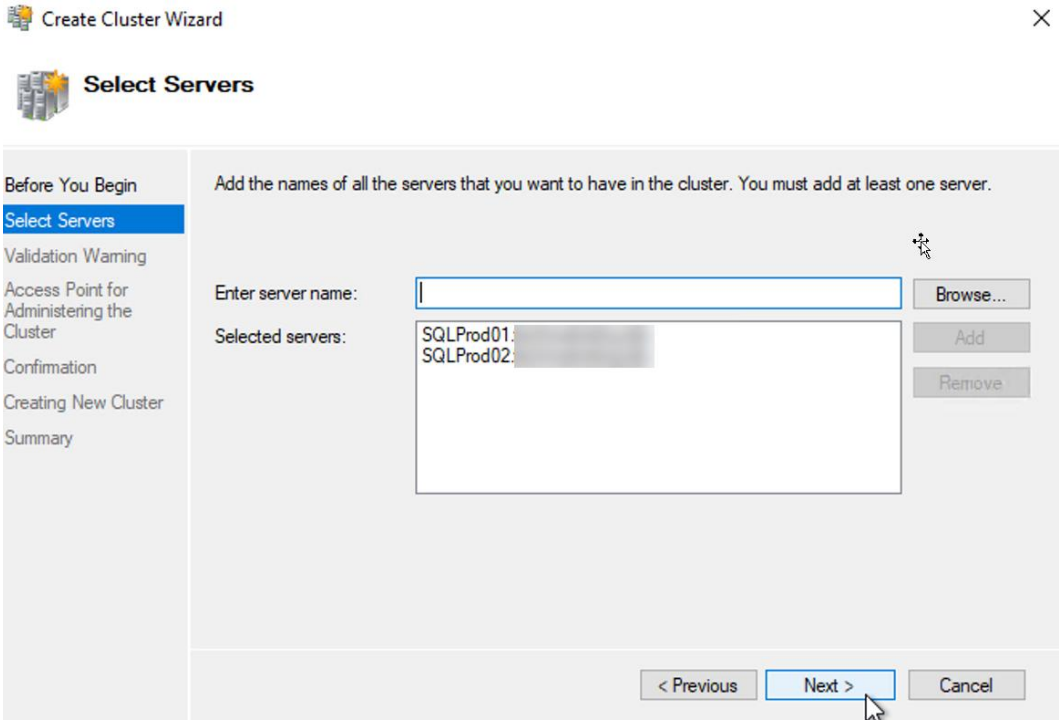
   PowerShell with administrator rights can also be used to add the failover clustering feature by running the following command:

```
Install-WindowsFeature -Name Failover-Clustering -IncludeManagementTools -Restart
```

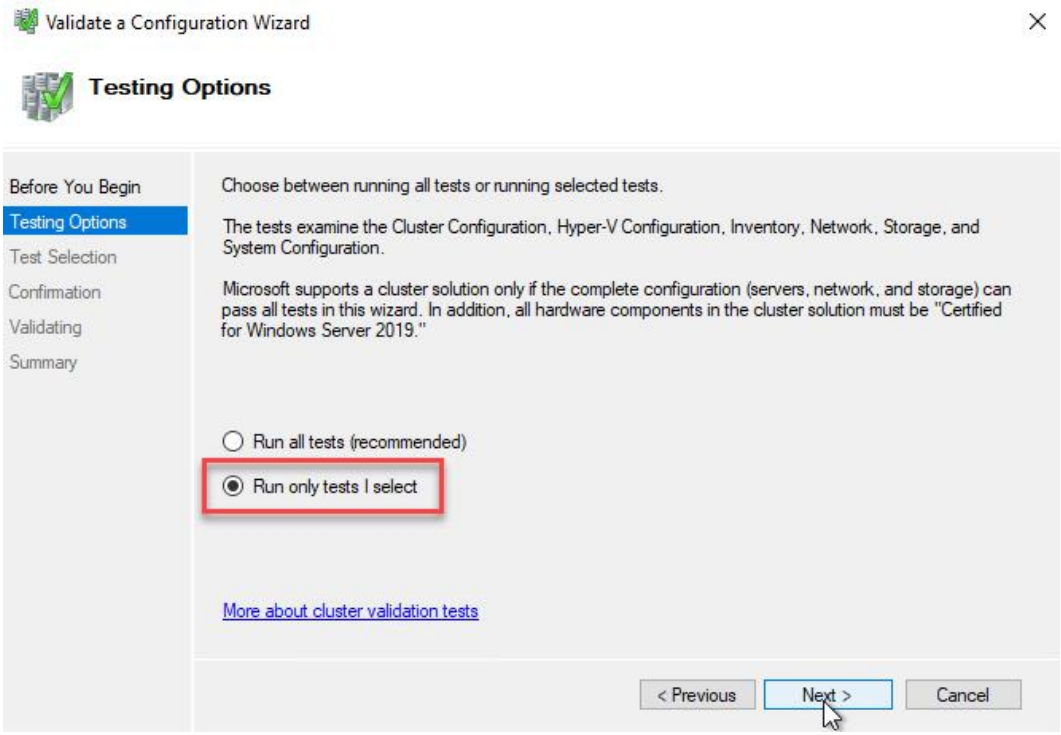
2. Validate the configuration and create Windows failover cluster by adding participating servers.

   If you are planning to use SQL Server 2016, do not use the user interface to create Windows failover cluster because this will cause dictionary key errors. Instead, use the following PowerShell command to create and validate Windows failover cluster.

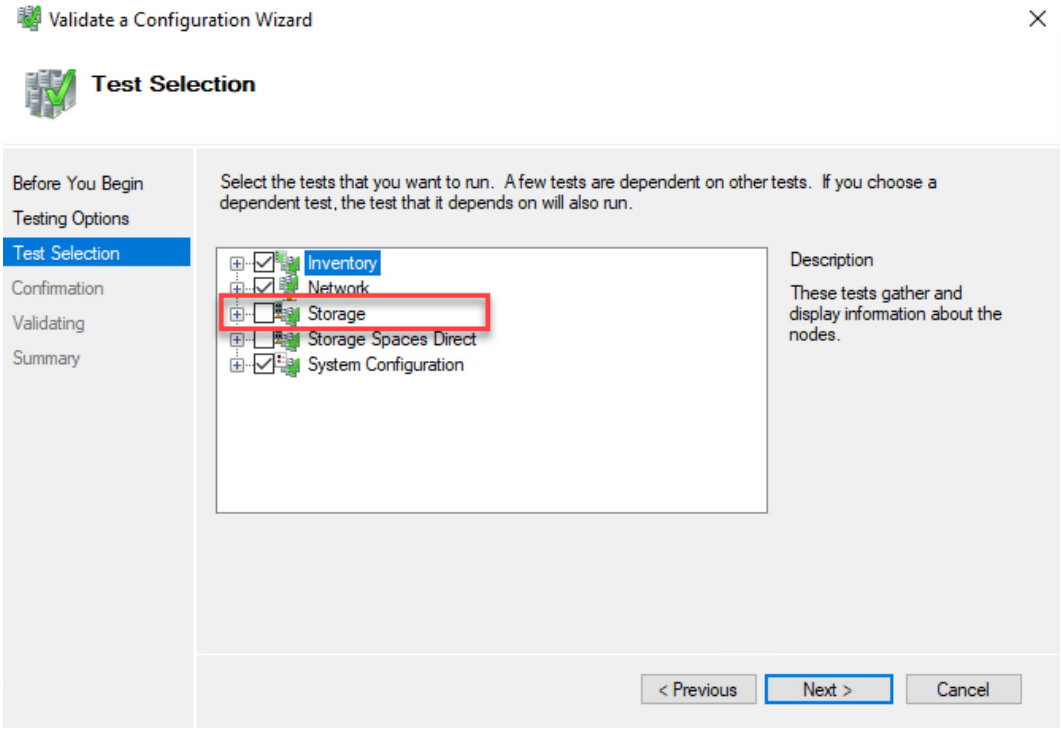
```
$ClusterName = "SQLProdCluster"
$node1 = "SQLProd01"
$node2 = "SQLProd02"
$sqlvip = "10.1.1.68"
$quorum = '\\gcpsmb\quorum' #This is CVS path that create with GCP Portal

New-Cluster -Name $ClusterName -Node $node1,$node2 -NoStorage -StaticAddress $sqlvip -
managementpointnetworktype singleton
Set-ClusterQuorum -FileShareWitness $quorum -Credential $(Get-Credential)
Test-Cluster -Node $node1, $node2
```



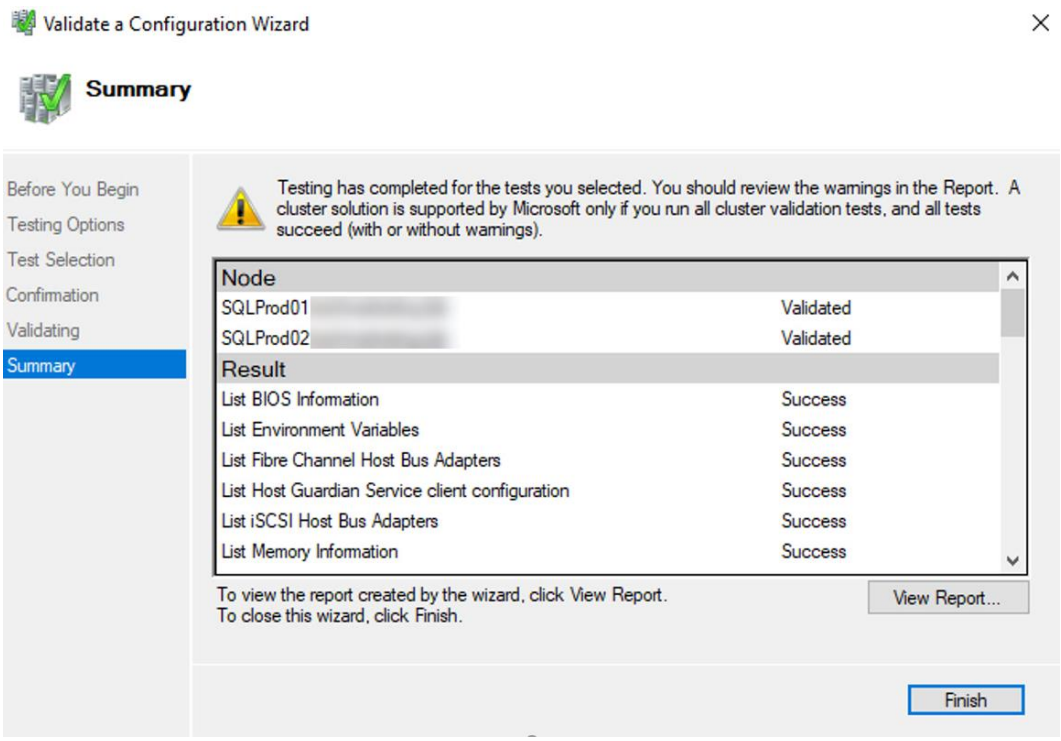
3. In the Validate a Configuration Wizard, under Testing Options, select Run Only Tests I Select. Click Next.



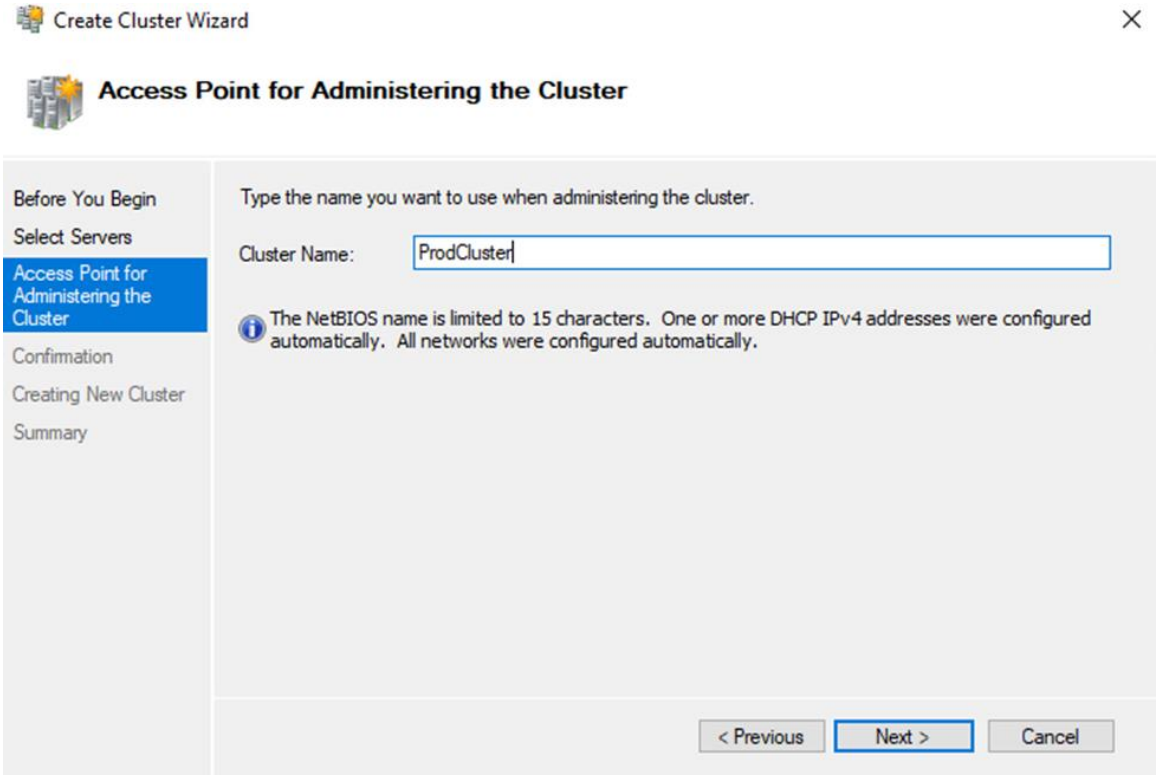
4. Under Test Selection, clear the Storage option because local storage is not going to be used.



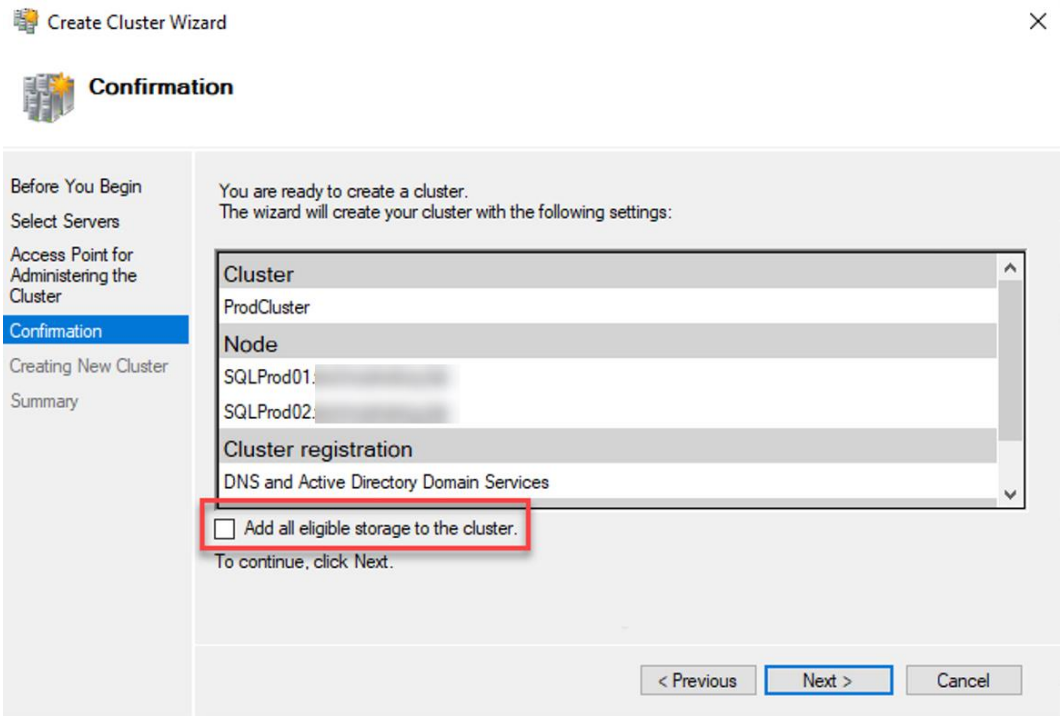
5. Under Summary, click Finish to complete validation configuration.



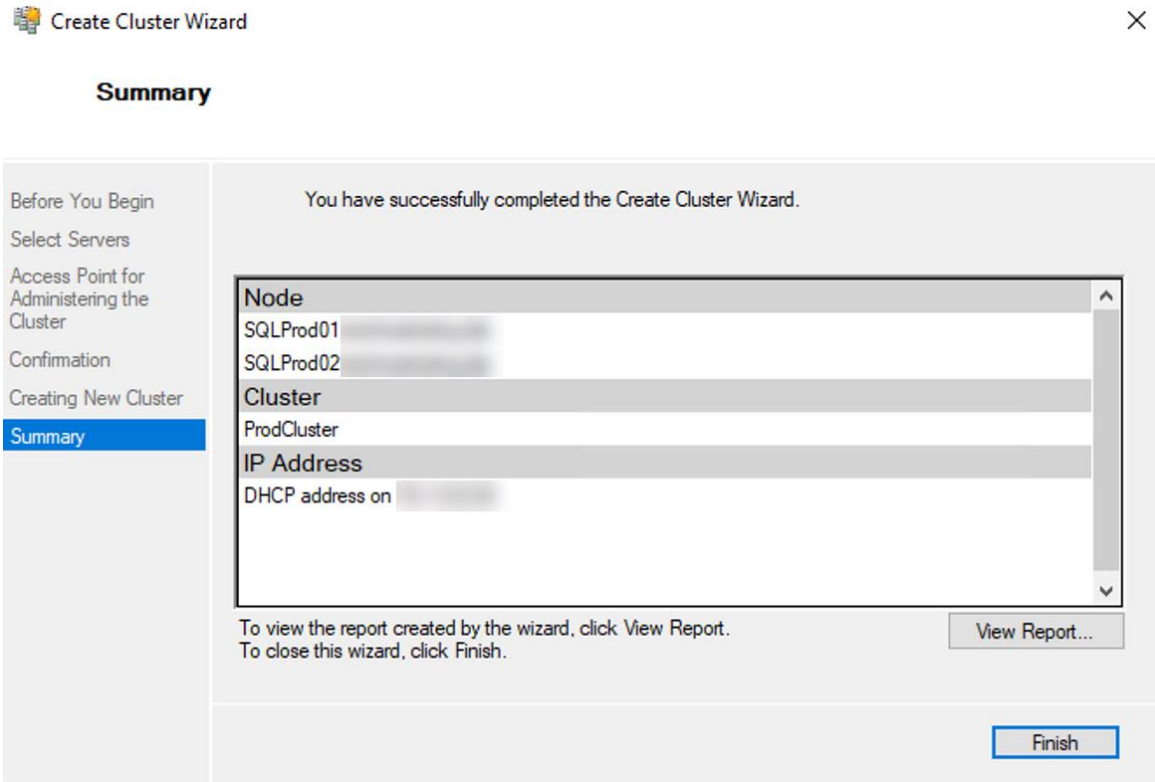
6. In the Create Cluster Wizard, under Access Point for Administering a Cluster, select Create Cluster and provide a cluster name. Click Next.



7. Under Confirmation, clear the Add All Eligible Storage to the Cluster option and click Next.



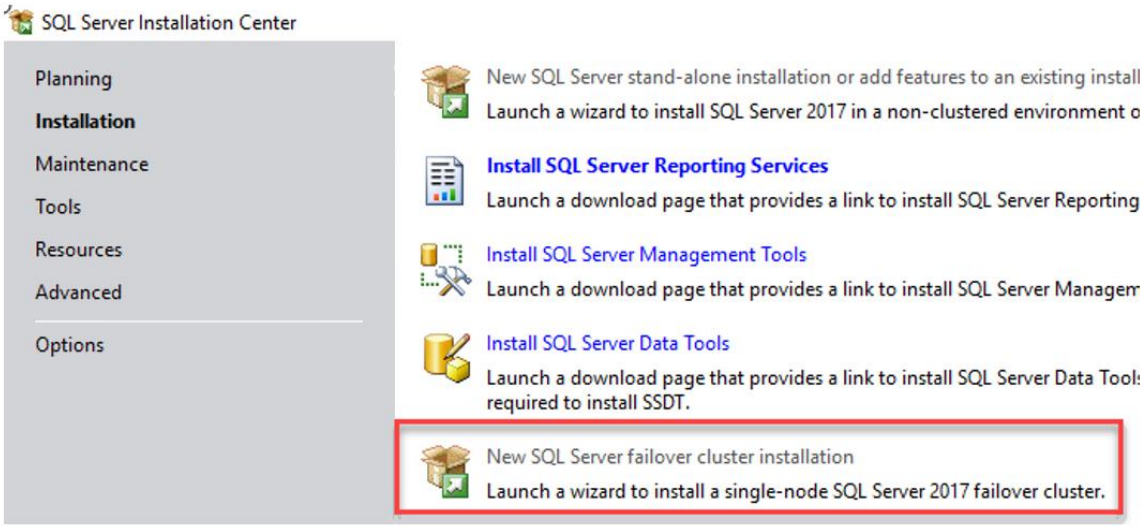
8. Under Summary, click Finish.



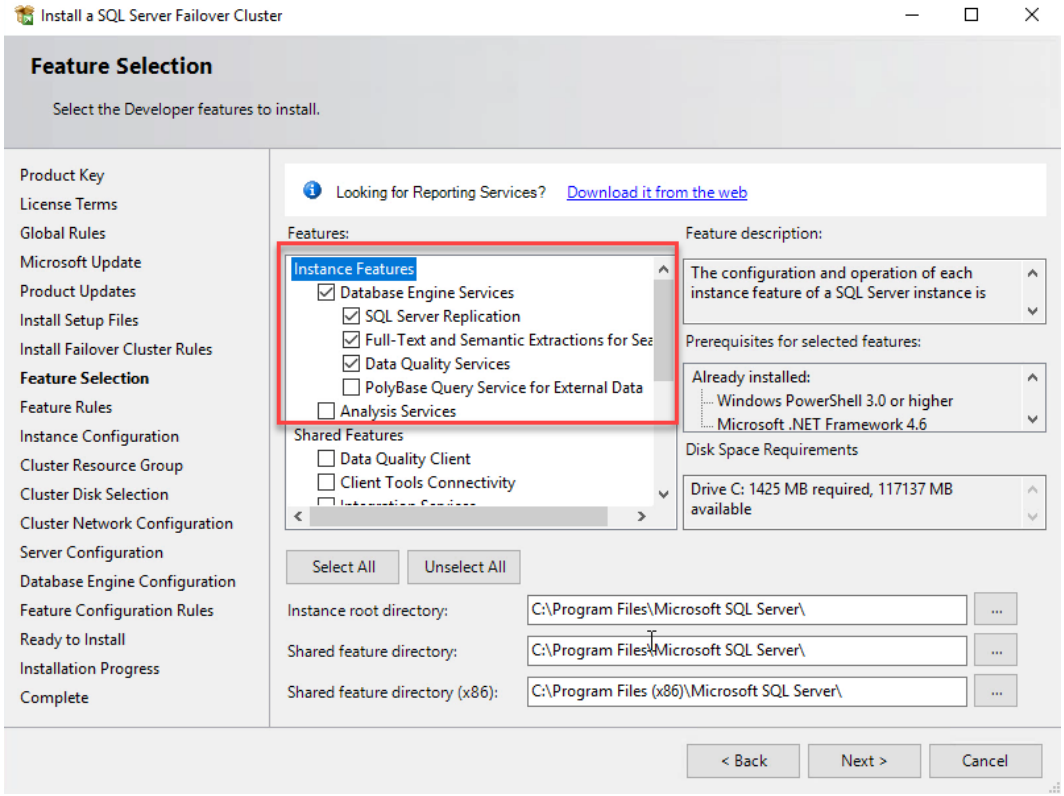
Install a new SQL Server failover cluster

To install a new SQL Server failover cluster, complete the following steps:

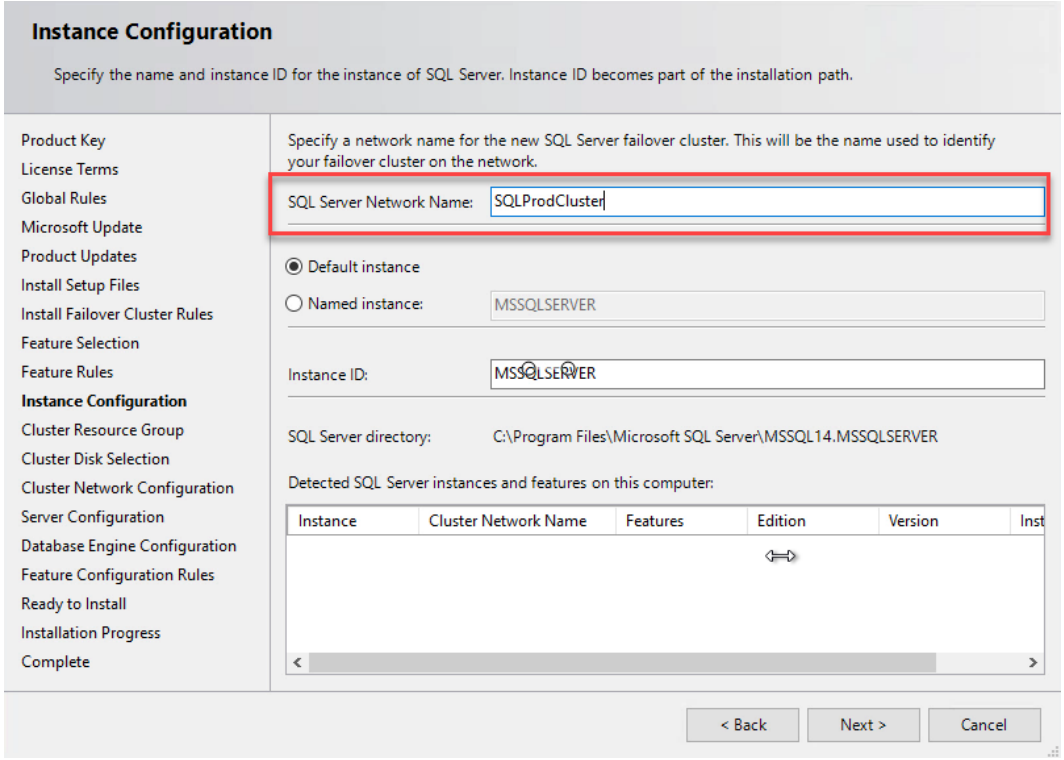
1. In SQL Server Installation Center, select Installation and New SQL Server Failover Cluster Installation.



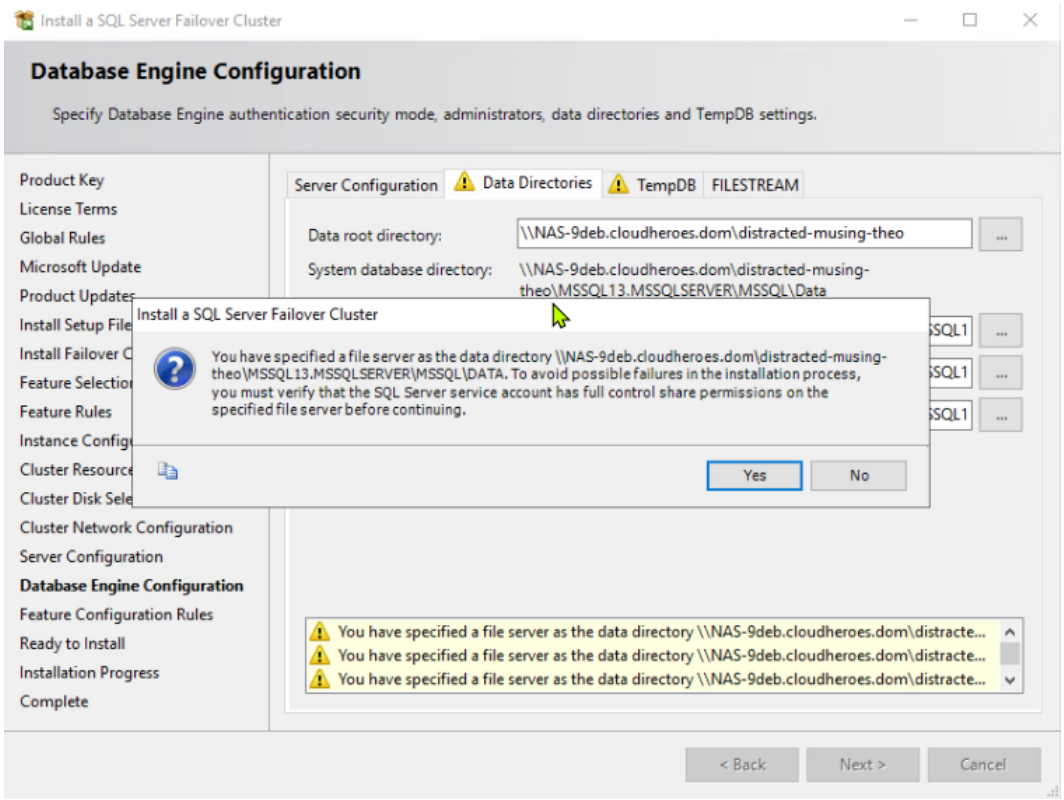
2. Provide the product key and accept the license terms.
3. Under Feature Selection, select Database Engine Services.



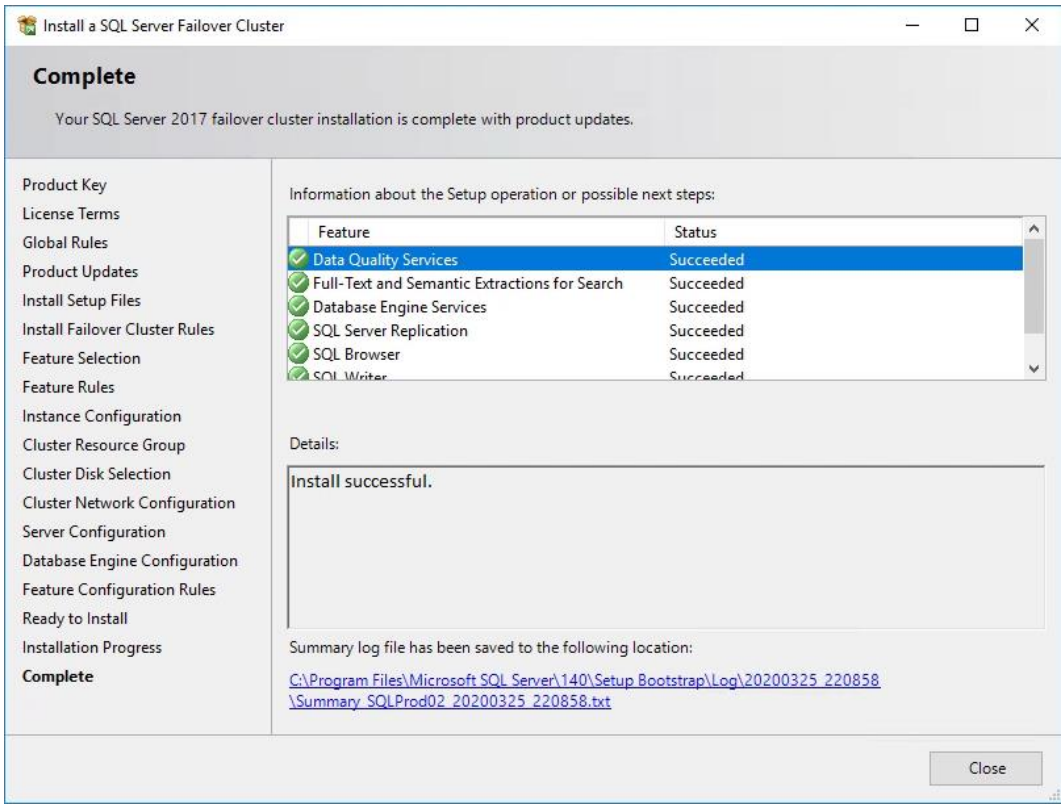
4. Provide the SQL Server network name and click Next.



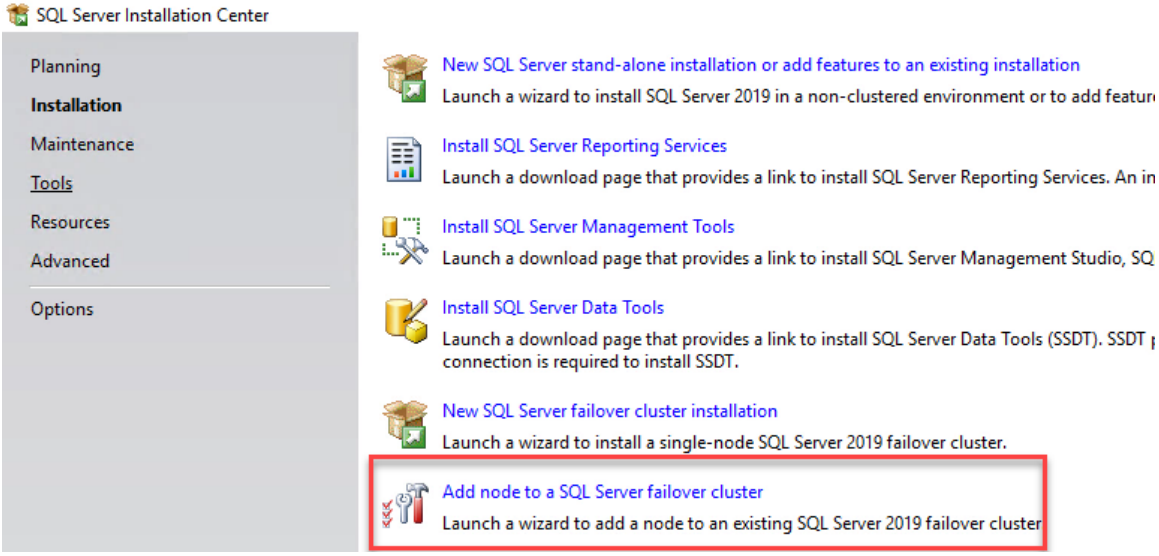
5. In Database Engine Configuration, select the Data Directories tab and provide the data root directory with Cloud Volumes Service SMB volume paths.
6. In the information dialog box, click Next and then Yes.



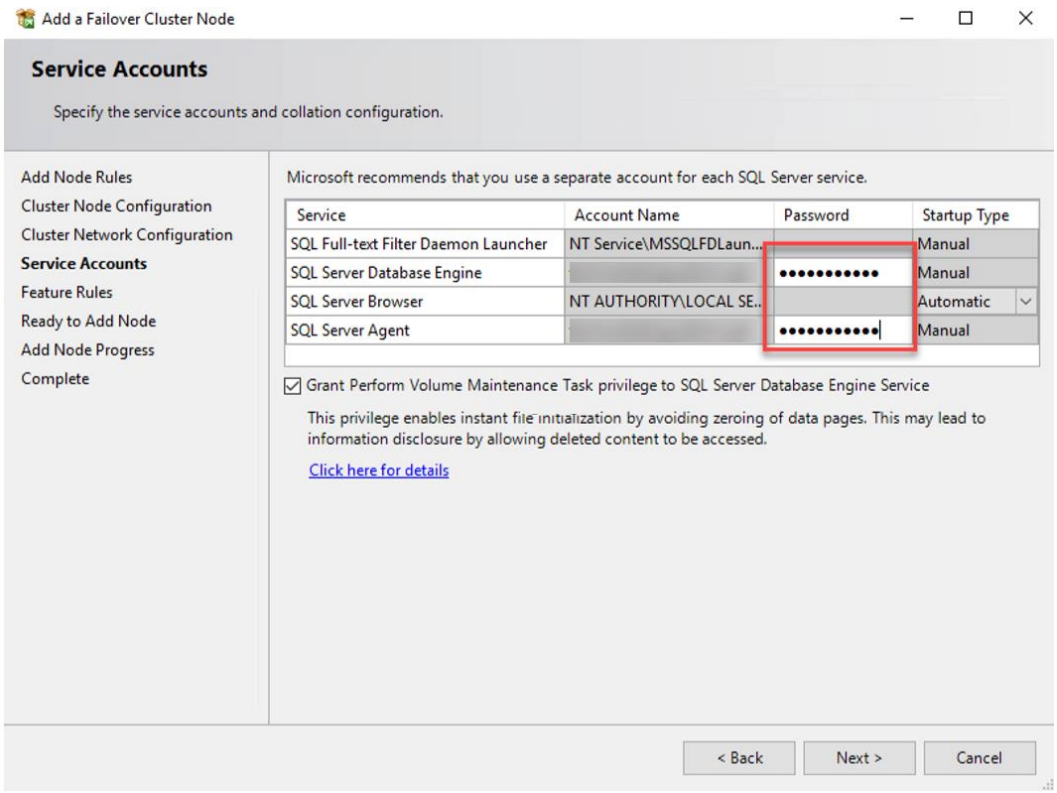
7. Finish the installation and click Close.



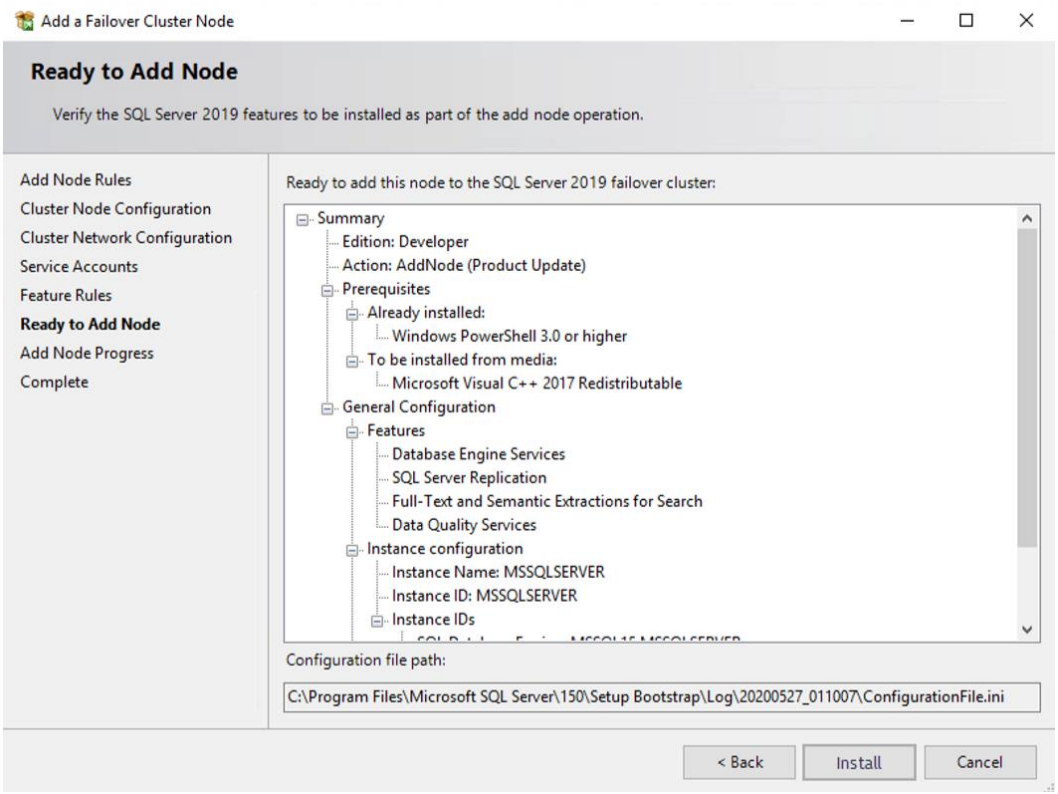
8. To install SQL Server in the second node, in the SQL Server Installation Center, select Installation then select Add Node to a SQL Server Failover Cluster .



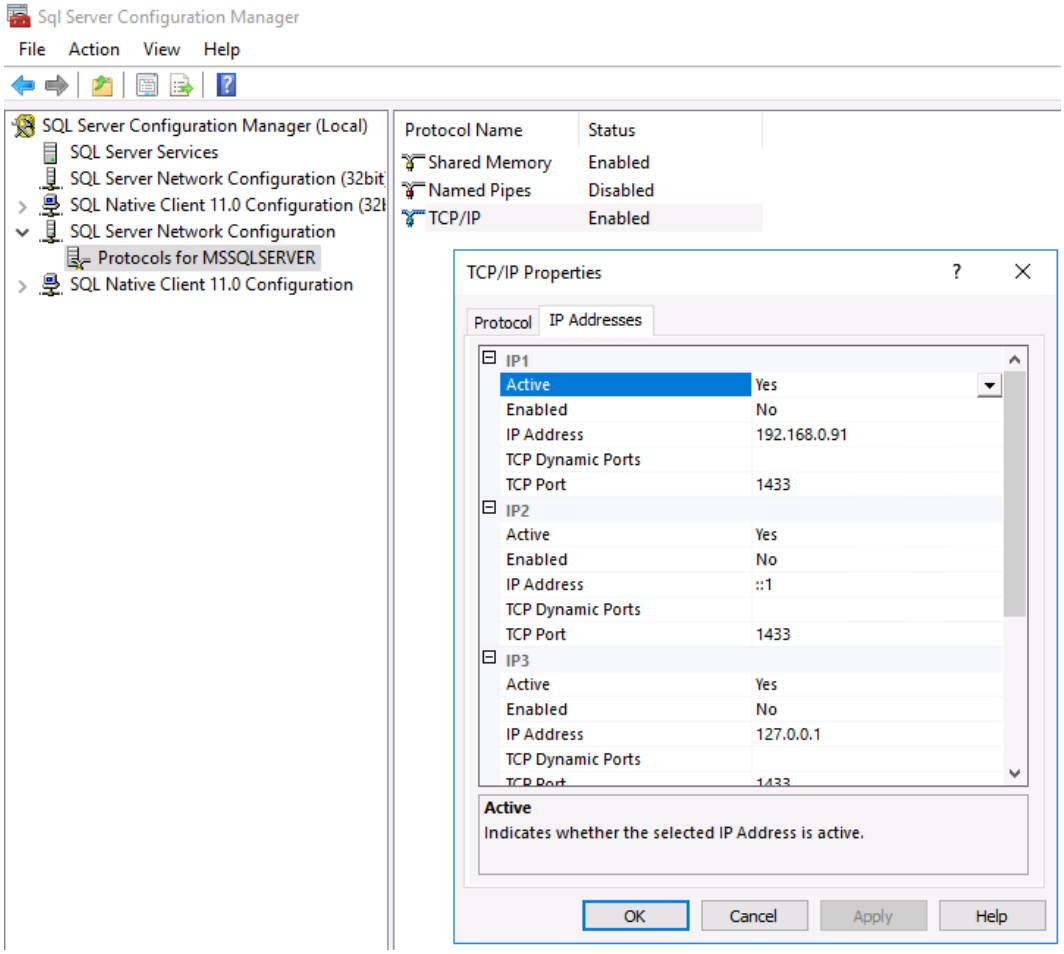
9. Provide the product key and accept the license terms.
10. Provide the passwords for the SQL Server database engine and the SQL Server agent service accounts. Click Next.



11. To finish the installation, click Install.

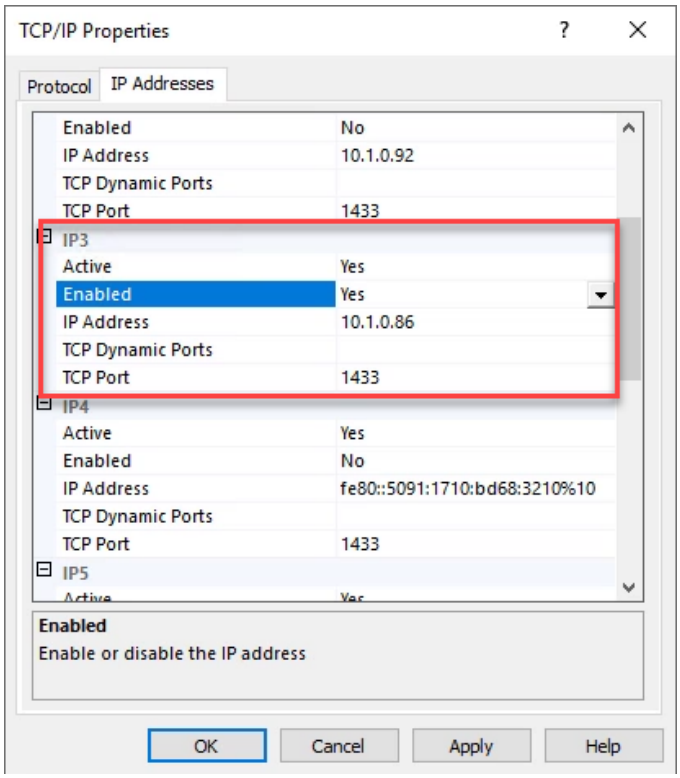


12. To enable virtual IP for SQL Server FCI, start SQL Server Configuration Manager.
13. Select SQL Server Network Configuration and then select Protocols for MSSQLSERVER.
14. Right-click TCP/IP and then select Properties.

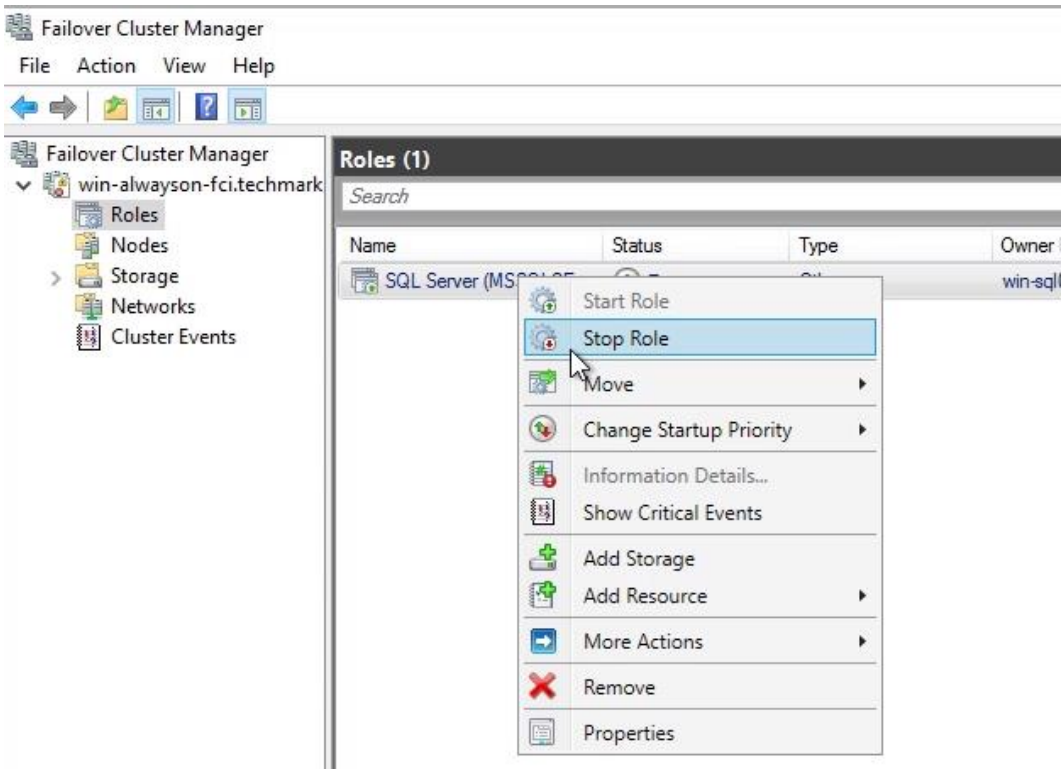


15. Select IP Addresses and enable IP3 for virtual IP Address and click OK.

    Note: Make sure that the port is set to 1433 or the default port of SQL Server during installation.



16. In Failover Cluster Manager, restart SQL Server by selecting Roles.
17. Restart the SQL Server Cluster Role by selecting Stop Role and then Start Role.



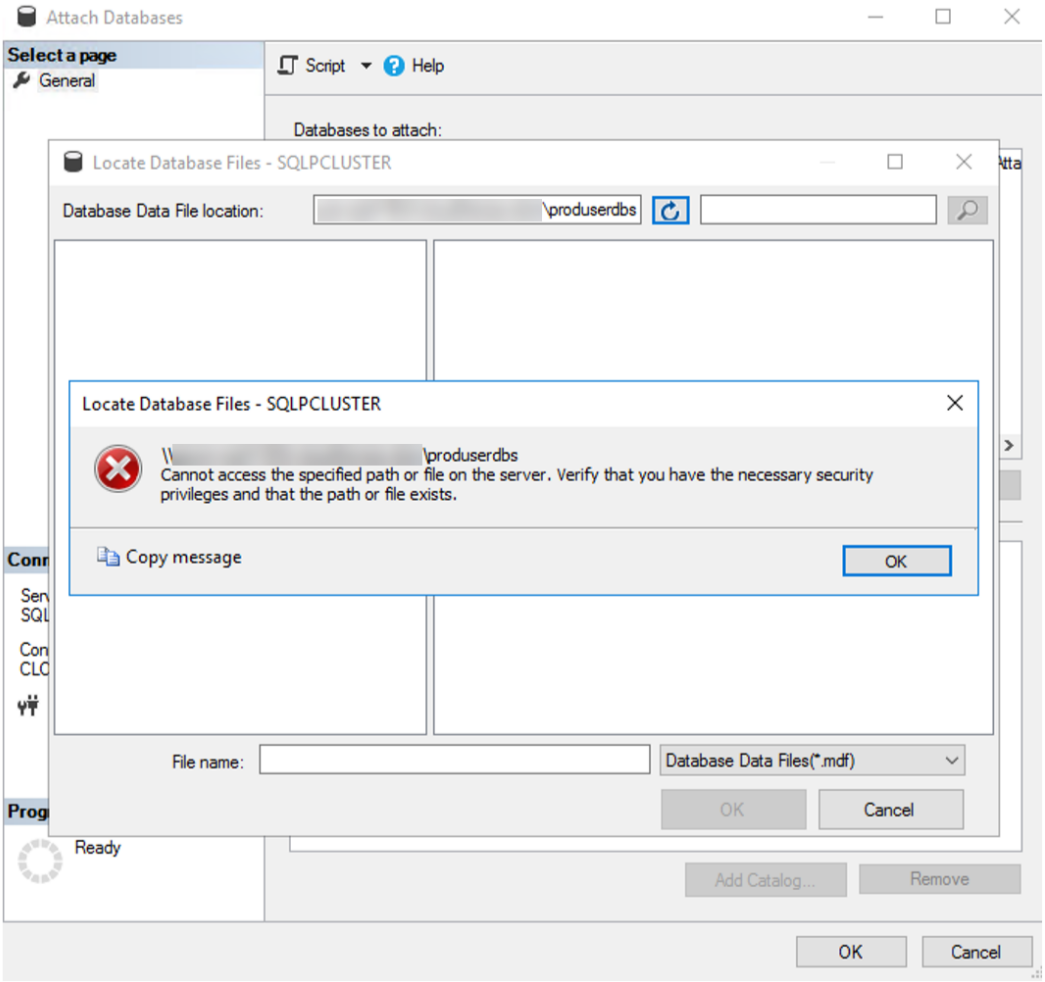
Note: You must deploy an internal load balancer that uses a health check to ensure that traffic is directed to the active node of the Windows Failover Cluster.

Commonly known errors

If you use T-SQL to attach or restore databases, a message like this might appear in query result and log in SQL Server log.

```
Msg 5120, Level 16, State 101, Line 1 Unable to open the physical file
"\\servername\sharename\filename.mdf". Operating system error 5: (Access is denied.).
```

If you are using SQL Server Management Studio, an error message like this might occur.



To resolve, follow these steps:

1. Ensure that the SQL Server service account has full access to the Cloud Volumes Service volumes.
2. Ensure that SQL Server service has full access to the data and log files.
3. In SQL Server Configuration Manager (for information about 1802, see Trace Flags), use this startup option to turn on trace flag 1802. For more information about how to change the startup parameters, see Database Engine Service Startup Options.
4. To reattach the database instead of SQL Server Management Studio, run the following T-SQL command:

```
exec sp_attach_db DatabaseName, '\\Network-attached storage_Path\DatabaseMDFFile.mdf',
'\\Network-attached storage_Path\DatabaseLDFFile.ldf'
go
```

Where to find additional information

To learn more about the information that is described in this document, review the following documents and/or websites:

- NetApp Product Documentation
  https://www.netapp.com/support-and-training/documentation/

Version history

Version	Date	Document version history
Version 1.0	August 2021	Initial release.

Refer to the Interoperability Matrix Tool (IMT) on the NetApp Support site to validate that the exact product and feature versions described in this document are supported for your specific environment. The NetApp IMT defines the product components and versions that can be used to construct configurations that are supported by NetApp. Specific results depend on each customer's installation in accordance with published specifications.

TR-4898-0821