Zerto
a Hewlett Packard Enterprise company

Best Practices for Protecting Microsoft SQL Server Failover Cluster Instances with RDMs

Replication, Failover, and Failback of Microsoft SQL Server Failover Cluster Instances with a Zerto Managed Service Provider

Version 1.0

Co-authored with Assurestor

Assurestor

Zerto
a Hewlett Packard Enterprise company

Contents

General Considerations 3

Failover Testing 3

How to Perform a Live Failover 4

Post-Failover Configuration 4

Automating Post-Failover Configuration by Script 5

Rolling Back Post-Failover Configuration 5

How to Perform a Failback 5

Limitations and Considerations 7

Conclusion 7

Zerto, a Hewlett Packard Enterprise company, supports continuous data protection (CDP) for Microsoft SQL Servers (MSSQLs), including Failover Cluster Instances (FCIs). When protecting an MSSQL Failover Cluster with Zerto, the key consideration is consistency of the database raw device mappings (RDMs) and cluster itself. In this paper, we'll look at the best practices for replication, failover testing, live failovers, and failback. In particular, we'll focus on a scenario leveraging a Zerto managed service provider (MSP) partner as part of disaster recovery as a service (DRaaS).

General Considerations

To start, note that only the Primary Active Node in an Active/Passive Cluster should be protected by Zerto.

1. When protecting a Failover Cluster to DRaaS through a Zerto Cloud Connector, the cluster witness must be a file share witness. Disk-based witnesses are NOT supported.
2. A failover operation will recover the protected active node to the recovery site. However, because the recovered node will be using non-shared VMDKs), you will need to manually intervene to evict the disks from the cluster instance, which allows them to come online as normal disks. After this, the cluster roles should start normally, bringing the SQL instance(s) and databases online.
3. A failback operation could be complex depending on the availability of the original RDM pointer VMDKs and LUNs.
4. If the active node role is switched to a non–Zerto-protected node, then any changes made to the RDMs are not replicated. When you move the active node role back to the Zerto-protected node, the Virtual Protection Group (VPG) will be in an inconsistent state because the target RDMs contain data Zerto did not replicate. To ensure the VPG is in a consistent state, you will need to perform a Force Sync operation.
5. The Force Sync operation scans both the source and target RDMs/VMDKs, then replicates any changes and inconsistencies it finds. A manual Force Sync operation maintains cluster consistency during maintenance. For example, during cluster maintenance, when the administrator changes the active node role back to the Zerto protected node, their final action should be to Force Sync via the Zerto GUI or API/script.

> **Note:** Depending on your settings for journal history, your settings for journal hard limit, and the duration of the Force Sync, you can lose journal checkpoints during a Force Sync in some limited circumstances. However, in accordance with Healthy Journal Protocol, Zerto will always attempt to maintain at least some checkpoints to ensure recovery is possible. If you are concerned about the Force Sync and its possible implications, please contact Support for guidance before you perform this operation.

Failover Testing

To successfully perform a nondisruptive failover test of a Microsoft SQL VM configured as an RDM-based FCI, the following need to be online in the failover test isolated network: Active Directory, DNS, and the file share witness. Therefore, Zerto recommends protecting an Active Directory Domain Controller (configured as a Global Catalog), the primary or secondary DNS server, and the file share witness server for the SQL Server VM. Zerto can then easily bring online an updated copy of Active Directory for failover testing.

Note: The Active Directory VM should never be recovered to previous points in time in a production/live failover. Zerto recommends placing the Active Directory VM in its own VPG and assigning both failover and failover test network adapters in the VM to connect to an isolated test network. Zerto recommends adhering to Microsoft best practices for Active Directory for production/live failovers.

Note: When booting the Active Directory in an isolated test network, a minimum window of five minutes is required for Active Directory services to come fully online to allow the cluster services to start.

How to Perform a Live Failover

1. Use these steps to successfully perform a live failover of a Microsoft SQL VM configured as an RDM-based FCI:
2. Ensure an Active Directory server configured as a Global Catalog is available.
3. Ensure the active node's primary and secondary DNS servers are available.
4. Ensure the cluster file share witness server is available.
5. If the passive node is powered on, shut it down.
6. Start the failover live workflow.
7. If the original site remains available, ensure that Reverse Protection is selected and configured correctly. If the original RDM VMDK pointer disks and LUNs still exist, Zerto will use these as seed disks, triggering a Delta Sync on commit.
8. If the original site is unavailable, fail over without Reverse Protection.
9. Complete post-failover configuration.
10. Commit failover.

Post-Failover Configuration

As only the active node is recovered, this will bring the cluster up as a non-clustered instance with no high availability (HA) functionality. You will need to manually intervene to bring the cluster online.

1. Using Failover Cluster Manager, evict the original RDM disks (which are now non-shared VMDKs) from the cluster, releasing the disk reservation and any role dependencies on the disks.
2. Using Disk Management, bring any offline disk(s) online in read/write mode.
3. Confirm you can now see the original RDM disk partition(s) and data files.
4. Bring the MSSQL Cluster role online.
5. Verify that the role starts correctly and that you can access the MSSQL instance(s) and databases as expected.

Note: Zerto can automatically change the IP address of VMs as part of a failover or failover test operation. However, if an MSSQL cluster requires a new IP address on the target site, this feature should not be used. This is due to issues with clusters and IP changes that can require manual intervention as part of a failover operation, which significantly increases RTOs and complexity.

Automating Post-Failover Configuration by Script

The following script can be used on the recovered active node to automate the required Microsoft Cluster Server (MSCS) disk changes, allowing the cluster role(s) to be started.

```
# Remove Disks from MSCS
Get-ClusterResource | where {$_.ResourceType.Name -eq "Physical Disk"} | Remove-ClusterResource -Force

#Set Offline disks to Online
Get-Disk | Where-Object IsOffline -Eq $True | Set-Disk -IsOffline $False

#Set ReadOnly disks to ReadWrite
Get-Disk | Where-Object IsReadOnly -Eq $True | Set-Disk -IsReadOnly $False

#Start Cluster Role
Get-ClusterGroup | Start-ClusterGroup
```

Rolling Back Post-Failover Configuration

To roll back the post-configuration changes, you must add the cluster-compatible disks back to the cluster and MSSQL role through the following steps:

1. Use PowerShell to add individual disks back into the cluster (where "x" is the disk number as shown in Disk Management).
   `Get-Disk -Number x | Add-ClusterDisk`
2. Use PowerShell to add all offline disks back into the cluster.
   `Get-Disk | Where-Object IsOffline -Eq $True | Add-ClusterDisk`
3. Use Failover Cluster Manager to Add Storage, selecting the cluster disks to the MSSQL role.

How to Perform a Failback

Failback should always be performed using the Move workflow to ensure the VM is failed back in an application-consistent state. The failback process differs based on two key scenarios: whether the original site is still up and accessible or down and inaccessible.

If the original site remained available, including the original RDM VMDK pointer disks during failover, the original VPG should still exist now, replicating the recovered VM back to the original site. To fail back with an available original site:

1. Verify the VPG is in a healthy sync state.
2. Start the Move workflow with Reverse Protection.
3. Roll back the post-failover configuration.
4. Confirm you can now see the original RDM disk partition(s) and data files.
5. Bring the MSSQL Cluster role online.
6. Verify the role starts correctly and that you can access the MSSQL instance(s) and databases as expected.
7. Commit Move.
8. If the passive node is available, power it on. If the passive node is unavailable, build a new node and add it to the cluster.
9. Verify that the MSCS is healthy.

If the original site and/or original RDM VMDK pointer disks were unavailable during failover, a new VPG will need to be created to replicate the recovered VM from the DRaaS environment back to the end user's site. To fail back with an unavailable original site:

1. Verify that the DRaaS environment is healthy and connected to the end user site.
2. Create a new VPG to replicate the recovered VM to the end user site.
3. Verify that the VPG is in Sync.
4. Start the Move workflow.
5. Verify that the Moved VM powers on.
6. Verify that the cluster role starts correctly and that you can access the MSSQL instance(s) and databases as expected.
7. Commit the Move.

> **Note:** Because the original RDM volumes were unavailable during the failover commit, disks can only be replicated back to the end user's site as standalone VMDKs. To re-establish the original MSCS MSQL Cluster using RDMs, the end user will need to build a new MSCS MSQL Cluster with new RDM disks, and then migrate the database(s) from the failed back VM to the new MSCS MSQL cluster nodes.

Limitations and Considerations

Zerto MSSQL Failover Cluster support to DRaaS is not compatible with the following:

1. Active-Active cluster—all SQL instances must run on the same node (Active-Passive).
2. Clusters that use disk-based witness for the Quorum.
3. Clusters that use Cluster Shared Volumes (CSVs).
4. Protecting Cluster VMs using iSCSI in-guest initiators to access shared cluster disks.
5. Odd block-sized RDMs (see https://help.zerto.com/kb/000002975).

Conclusion

Zerto can ensure robust protection and recovery of MSSQL Server Failover Cluster Instances that are using RDMs. Special attention must be paid to the various Zerto operations to ensure successful replication and database consistency. Additional resources on protecting SQL databases, RDMs, and using Zerto with an MSP for DRaaS are available in the Zerto technical documentation available at https://help.zerto.com.

RITM0102524