

SQL Server Failover Clustering is a high-availability solution that allows SQL Server instances to automatically switch to another node in the event of a failure. However, there can be various issues that may arise in a SQL Server failover cluster environment.

Below is a detailed look at frequent SQL Failover Cluster issues and their resolutions.

1. Cluster Service Fails to Start

Issue: The Cluster Service fails to start on one or more nodes, causing SQL Server to become unavailable.

Causes:

- Missing or corrupted cluster configuration files.
- DNS issues causing cluster name resolution failures.
- Insufficient permissions for the Cluster Service account.
- A recent cluster or node crash causing corrupted registry keys or cluster database issues.

Resolution:

- **Check Permissions:** Ensure the Cluster Service account has the necessary permissions in Active Directory and on all cluster nodes. It should be a member of the local **Administrators** group on each node.
- **DNS Resolution:** Ensure the cluster name and IP address are properly registered in DNS, and that all nodes can resolve the cluster name.
- **Repair Cluster Configuration:** Run the following command to repair the cluster service:

```
cluster /fixquorum
```

Check Cluster Logs:

Review the Windows Cluster logs (**C:\Windows\Cluster\Reports**) for detailed errors. Check event logs for related errors in the **FailoverClustering** source.

Verification:

- Ensure the Cluster Service starts successfully by using **Failover Cluster Manager** or running the following command:

```
net start clussvc
```

2. SQL Server Service Fails to Come Online in the Cluster

Issue: The SQL Server service fails to come online after a failover or cluster node reboot.

Causes:

- The SQL Server service account does not have permissions to bring the SQL Server instance online.
- Incorrect SQL Server network configuration, such as conflicting ports.
- Corrupt SQL Server database files or log files.
- Dependencies (e.g., disks, IP addresses, or network names) are not online.

Resolution:

- **Permissions:** Verify that the SQL Server service account has permissions on the clustered disks and the SQL Server binaries.
- **Check Network Configuration:** Ensure that no other services are using the same port that SQL Server is configured to use. Modify the SQL Server port if necessary.
- **Check Database Files:** Ensure that all required database files and log files are present and accessible on the shared storage. Check SQL Server error logs for any file corruption or access issues.
- **Resolve Dependency Issues:** Ensure that all dependent resources (IP addresses, disks, etc.) are online before SQL Server tries to come online.

Verification:

- Use the **Failover Cluster Manager** to check the status of the SQL Server instance and dependent resources.
- Review the SQL Server error logs and Cluster Logs for specific errors related to the failed startup.

3. Cluster Network Name Fails to Come Online

Issue: The network name for the SQL Server failover cluster instance fails to come online, causing clients to be unable to connect.

Causes:

- Active Directory (AD) object for the cluster network name is missing or corrupt.
- DNS registration issues for the network name.
- Insufficient permissions for the SQL Server cluster computer object to create/update DNS records.

Resolution:

- **Repair AD Object:** If the Active Directory object for the SQL Server cluster network name is missing or corrupt, recreate it. Ensure that the cluster computer object has the necessary permissions in AD to update DNS records.
- **DNS Registration:** Verify that the SQL Server cluster network name is correctly registered in DNS. You can use the following command to manually register the cluster name in DNS:

```
ipconfig /registerdns
```

- **Permissions:** Ensure that the SQL Server cluster service account has the necessary permissions to create and update DNS records in the domain. Use the following steps:
 1. In **AD Users and Computers**, check the **Security** settings for the cluster computer object.
 2. Ensure that the service account has **Full Control** or the **Create Computer Objects** permission in AD.

Verification:

- Check the **Failover Cluster Manager** to ensure that the network name resource is online.
- Use `nslookup` to verify that the cluster network name is correctly resolved to its IP address.

4. Quorum Configuration Issues

Issue: Cluster nodes lose quorum, causing the cluster to fail or operate in a degraded state.

Causes:

- Insufficient quorum votes due to node failures or misconfigured quorum models.
- The quorum witness (disk or file share) is unavailable or incorrectly configured.
- Network partitioning between nodes causes a lack of quorum.

Resolution:

- **Check Quorum Configuration:** Ensure that the cluster is using the appropriate quorum model based on the number of nodes. You can use either **Node Majority**, **Node and Disk Majority**, or **Node and File Share Majority** depending on your setup. Use the following command to check and configure quorum:

```
cluster.exe /quorum:<QuorumModel> /witness:<WitnessType>
```

- **Verify Witness Availability:** If you are using a quorum witness (disk or file share), ensure that it is online and accessible by all nodes.
- **Resolve Network Issues:** Ensure there are no network connectivity issues between cluster nodes, as network partitioning can prevent quorum from being reached.

Verification:

- Use the **Failover Cluster Manager** to check the current quorum configuration and status.
- Run the following command to verify quorum status:

```
cluster.exe /quorum /status
```

5. Failover Takes Too Long or Fails

Issue: SQL Server failover is taking too long or fails altogether, causing downtime.

Causes:

- The SQL Server instance is taking too long to shut down on the current active node.
- Resource dependencies (e.g., disk, IP address, network name) are slow to fail over.
- Heavy database activity (large open transactions or checkpoints) during failover.

Resolution:

- **Check SQL Server Shutdown:** Ensure SQL Server is able to shut down cleanly during failover. Investigate the SQL Server logs to identify any long-running queries or open transactions that are causing delays.
- **Reduce Dependency Failover Time:** Review the **Failover Threshold** and **Retry Interval** settings for critical resources (like disks and IP addresses). Consider lowering the timeouts for faster failover.
- **Use Optimized Failover Settings:** Configure the SQL Server resource properties for faster failover, using the following steps:
 1. In **Failover Cluster Manager**, right-click the SQL Server resource, select **Properties**.
 2. In the **Advanced Policies** tab, reduce the **Maximum Failover Attempts** and **Failover Threshold** values if necessary.

Verification:

- Monitor the failover process using **Failover Cluster Manager** and check for bottlenecks in resource transition.
- Review the SQL Server logs for any errors or long-running queries causing delays.

6. Cluster Disk Resource Fails to Come Online

Issue: The disk resource associated with the SQL Server instance fails to come online, preventing SQL Server from starting.

Causes:

- The shared disk storage (SAN or NAS) is not accessible to one or more cluster nodes.
- Corruption in the file system or partition of the disk.
- Disk drive letter conflict or incorrect mount point configuration.

Resolution:

- **Verify Disk Accessibility:** Ensure that the shared disk storage is accessible from all nodes. Use **Disk Management** to check the status of the disk on each node.
- **Check Disk Integrity:** Run `chkdsk` to check for file system corruption. If errors are found, attempt to repair the file system.
- **Drive Letter/Mount Point Conflicts:** Ensure that the disk is assigned the correct drive letter or mount point and that there are no conflicts with existing volumes on the nodes.

Verification:

- Use the **Failover Cluster Manager** to bring the disk resource online.
- Check the Windows Event Viewer for errors related to disk resource failures in the **System** log.

7. Cluster Network Partitioning

Issue: Network partitioning occurs between cluster nodes, leading to split-brain scenarios or failover issues.

Causes:

- Network failures or misconfigurations between the cluster nodes.
- Multiple network interfaces (NICs) on the nodes are incorrectly configured.
- Inconsistent network latency or bandwidth issues.

Resolution:

- **Network Configuration:** Ensure that the cluster network is correctly configured. For multi-NIC configurations, ensure that the correct NIC is used for cluster communication. Disable unnecessary network interfaces.
- **Check Firewall Settings:** Ensure that firewall settings allow communication between cluster nodes on the required ports (e.g., 3343 for cluster communication).
- **Network Redundancy:** Implement network redundancy by configuring multiple network paths between cluster nodes to prevent network partitioning.

Verification:

- Use the following command to check the status of cluster networks:

cluster.exe network /status

8. Cluster Resource Ownership Changes Unintentionally

Issue: Cluster resources, including SQL Server instances, change ownership between nodes unexpectedly, even when there is no failure.

Causes:

- Failover settings are too aggressive, leading to unnecessary failovers.
- Network or disk resource timeouts are set too low.
- The cluster nodes are experiencing transient connectivity or resource issues.

Resolution:

- **Adjust Failover Settings:** Check the **Failover Threshold** and **Retry Interval** settings for each cluster resource. Consider increasing these values to prevent unnecessary failovers.
- **Monitor Resource Usage:** Ensure that all cluster nodes have sufficient resources (CPU, memory, I/O) to handle the SQL Server workloads. If a node is under heavy load, it may trigger a resource failover.
- **Check Network/Disk Timeout:** Review the network and disk timeouts and adjust them based on the cluster's environment and load. Use the following command to modify resource timeouts:

```
cluster.exe resource <ResourceName> /prop LooksAlive=<TimeInSeconds>
```

Verification:

- Monitor the **Failover Cluster Manager** for unexpected resource ownership changes.
- Review the **Windows Event Viewer** for events related to resource failovers.

By addressing these common issues in a SQL Server Failover Cluster environment, you can ensure greater stability and availability for your SQL Server instances.

Regular monitoring, proper configuration, and timely troubleshooting are key to maintaining a healthy failover cluster.