Deploying an Exchange 2007 SP1 CCR Cluster on a Windows Server 2008 Failover Cluster

Back in October 2006 when Exchange Server 2007 still was a beta 2 product (at least to the general public), I wrote “Installing, Configuring and Testing an Exchange 2007 Cluster Continuous Replication (CCR) Based Mailbox Server” that walked you through how to deploy an Exchange 2007 clustered mailbox server (CMS) based on cluster continuous replication (CCR) cluster technology. Unlike a traditional Exchange cluster, a CCR cluster not only provides redundancy on the hardware level, but also provides redundancy on the storage level.

Since then clustered mailbox servers based on CCR clusters have become a very popular method for providing a cost effective high availability messaging solution that matches the service level agreement (SLA) in many of the Enterprise organizations around the world. Now that both Exchange Server 2007 Service Pack 1 (SP1) and Windows Server 2008 have been released to manufacturing (RTM), I thought it would be good timing to provide you with an article series that walks you through how to deploy an Exchange 2007 SP1 clustered mailbox server based on cluster continuous replication on Windows Server 2008, so you are prepared for those real world deployments that lie on the horizon.

Before we move on, let me first explain why it is advantageous to deploy an Exchange 2007 clustered mailbox server based on CCR cluster technology using Windows Server 2008 failover clusters instead of sticking with the well tested clustering functionality included with Windows Server 2003. First, Exchange Server 2007 SP1 can take advantage of the heavily improved failover cluster (formerly known as server cluster) component in Windows Server 2008. The Windows Server 2008 failover cluster component makes it even simpler to deploy CCR clusters and enhances the stability of the failover cluster solution as well as makes it more secure than in previous versions of Windows Server. In addition, because the new failover cluster included with Windows Server 2008 supports multiple subnets, unlike Exchange 2007 RTM, Exchange 2007 SP1 includes support for geographically dispersed clusters (GeoClusters) for failover across two subnets (although the nodes must still be in the same Active Directory site). Moreover, Exchange 2007 SP1 CCR clusters support both Internet Protocol version 4 (IPv4) as well as IPv6 and DHCP assigned IP addresses (both IPv4 and IPv6 although IPv4 is not recommended unless the DHCP server is set to grant leases for an unlimited length).

Note: Get a complete list of failover cluster improvements in Windows Server 2008.

Prerequisites
Before you can follow along with the steps provided in this article series, you must have a lab environment consisting of the following:

- 1 Windows 2003 SP1 (preferably SP2) or Windows Server 2008 Standard edition server configured as a domain controller in an Active directory forest with a domain functional level of at least 2000 Native.
- 1 Windows 2003 SP1 (preferably SP2) or Windows Server 2008 Standard edition server with the Exchange 2007 SP1 Client Access and Hub Transport server roles installed (optionally you can install these roles on the domain controller).
- 2 Windows Server 2008 Enterprise edition servers that will become the nodes in the failover cluster.

The 2 Windows Server 2008 Enterprise edition servers should be equipped with 2 network interfaces (one for the public network and one for the private heartbeat network) and at least 2 disks – 1 for the System partition, 1 for Exchange databases and optionally 1 for transaction log files (in a lab environment these could also be located together with the Exchange databases).

I used the 32-bit edition of Exchange 2007 SP1 and Windows Server 2008, which meant I could get along with allocating a lot less memory to each box (500 MB for the Domain Controller, 700 MB for the server where the Client Access and Hub Transport server roles are installed, and 700 MB for each failover cluster node) than would have been the case if I used 64-bit editions.

Although the above configuration recommendations are sufficient for a lab environment, please bear in mind that the 32-bit bit edition of Exchange 2007 is not supported in production environment and that best practice recommendations for Exchange 2007 CCR clusters in a production environment are very different from what I used here.

If you, like me, only have a limited set of test boxes it is highly recommended you deploy this lab environment using a virtualization solution. If you choose to use the 32-bit editions of Exchange 2007 and Windows Server 2008, you can use anything from Microsoft Virtual PC, Microsoft Virtual Server, VMWare workstation/server or a real enterprise virtualization platform such as VMWare ESX Server or Microsoft’s new HyperVisor (HyperV) included with Windows Server 2008. Just pick your favorite! :-)

### Configuring the Failover Cluster Nodes

Alright, when you have an Active Directory forest consisting of a domain controller and a member server with the Exchange 2007 Client Access and Hub Transport server roles installed we are ready to configue the two servers that will be the nodes in our failover cluster. The first thing we want to do when the servers have been turned on is to partition the disks and configure the network interfaces appropriately. To partition the disks, open the Windows 2008 Server Manager and expand Storage and then select Disk Management. Now right-click on each LUN that is offline and then select Online in the context menu as shown in Figure 1.
Figure 1: Bringing disks online

Right-click on one of the disks again and this time select Initialize Disk. In the Initialize Disk window, tick both disks and make sure MBR (Master Boot Record) is selected, unless you used disks larger than two Terabytes (TB) or use Itanium-based computers as nodes in the fail-cluster (both very unlikely!), and then click OK (Figure 2).

Figure 2: Initializing disks
The disks are now both online, and we can begin to partition them. To do so click on the unallocated disk space, then select New Simple Volume in the context menu (Figure 3).

![Figure 3: Creating the simple volumes](image)

In the New Simple Volume wizard, click Next. Specify the size of the new volume and click Next.

![Figure 4: Specifying the size of the volume](image)

We now need to assign a drive letter to the volume. When ready click Next (Figure 5).
It is time to label and format the volume. Do so and click Next (Figure 6).

Figure 5: Assigning a drive letter to the volume

Figure 6: Labeling and formatting the volume
Lastly click Finish to exit the wizard.

Perform the above steps for the second disk as well, but label it Logs (Figure 7).

![New Simple Volume Wizard](image)

**Figure 7:** Labeling and formatting the volume for the log files

Next step is to name the two network connections **Public** and **Private (heartbeat)** for the external and the internal network respectively as shown in **Figure 8**. The private network will be used only for the heartbeat between the two failover cluster nodes.
A general best practice when configuring the servers that will become nodes in a failover cluster is to change the binding order of the network interfaces. To do so click **Advanced > Advanced Settings**, if it is not already the case, make sure Public is listed first in the binding order list, then **Private (heartbeat)** and lastly **Remote Access Connections** as shown in **Figure 9**.
Figure 9: Binding order

When installing Windows Server 2008, Ipv6 is enabled by default, and although this Internet Protocol version is supported by Exchange 2007 SP1, we will not use it in this series. So you should untick Internet Protocol Version 6 (TCP/Ipv6) for both the Public and Private network interface as shown in Figure 10.
Figure 10: Disabling Ipv6 on the public network interface

In addition, it is recommended you disable Client for Microsoft Networks and File and Printer Sharing for Microsoft Networks on the Private network interface.

We can now begin to configure the IP settings for each network interface. Let us start with the public network interface (Figure 11).
Now move on and configure the IP settings for the private network interface. Notice you only should specify an IP address and a subnet mask for the private network interface as shown in Figure 12.
Figure 12: Configuring IP settings for the private network interface

Also make sure you untick Append parent suffixes of the primary DNS suffixes and Register this connection’s addresses in DNS (Figure 13).
Click the WINS tab and untick Enable LMHOSTS lookup and select Disable NetBIOS over TCP/IP (Figure 14).
Figure 14: Configuring WINS settings

With the network settings configured appropriately, let us move on and name each failover cluster node with meaningful names such as **CCRNODE1** and **CCRNODE2** or whatever naming scheme you want to use in your environment. When you have done so, add them as members in your Active Directory forest (Figure 15).
Finally reboot the servers for the changes to take effect.

The concludes part 1 of this 3 part article series revolving around how to deploy an Exchange 2007 SP1 CCR Cluster on a Windows Server 2008 Failover cluster. In part 2, which will be published soon here on MSExchange.org, we will form the Windows Server 2008 Failover Cluster as well as deploy the CCR based clustered mailbox server.

**Installing the necessary Windows Components**

Before we move on and try to install the Exchange Server 2007 Beta 2 bits, we need to make sure the required Windows components have been installed. All types of Exchange Server 2007 installations (no matter what server role we're talking about) needs the **Microsoft .NET Framework 2.0 component** installed.

**Note**

Since we’re installing the Mailbox Server role in the cluster, we also need to install the below IIS 6.0 components:

- Enable network COM+ access
- Internet Information Services
- World Wide Web Service

Note
Remember to install these components on both cluster nodes.

Configuring the Majority Node Set (MNS) Quorum with File Share Witness

I bet some of you are thinking: *What the heck is a Majority Node Set (MNS) Quorum with File Share Witness?* And I understand why as this is a completely new type of quorum model, which is made available by installing the update ([MS KB article 921181](https://support.microsoft.com/en-us/kb/921181)) mentioned in the beginning of this article series. The update makes it possible to make use of a file share witness that is external to the cluster as an additional "vote" to determine the status of the cluster in a two-node MNS quorum cluster deployment, which is a requirement in order to make use of the cluster continuous replication (or CCR in short) functionality in Exchange Server 2007.

The file share for this file share witness can be located on any type of Windows Server in your environment, but best practice is to use an Exchange 2007 Hub Transport Server in the Active
Directory server site containing the nodes in the respective cluster. We’ll also use a Hub Transport Server in this article series.

The first thing you need to do is to create the file share on the Hub Transport server. You can do this either via the CLI or by using the GUI. In this article we’ll do so using the GUI. So log on to the Hub Transport server with a domain admin account, then open Windows Explorer and create a new folder called MNS_FSQ_E2K7CLUSTER on the C: drive or wherever you want it to be created.

![Figure 28: Majority Node Set File Share Quorum folder](image)

Now take Properties for the newly created folder, and click Sharing.
Click **Permissions** and configure the share permissions so only the **Administrator** (or the Cluster Service Account if created) are allowed access to the share.
Figure 30: Share Permissions for Majority Node Set File Share Quorum folder

Click OK then select the Security tab.
Figure 31: Security permissions to the Majority Node Set File Share Quorum folder

Here you should give **Full Control** to the local administrator and the domain administrator account or cluster service account. Make sure you untick **Allow inheritable permissions from the parent to propagate to this object and all child objects** when doing so, then click **OK** twice and log off the server.

Back on **E2K7Node1** you should set the Majority Node Set Private Property attribute to point to the file share we just created, we do so by opening a command prompt then issuing the following command:

```
Cluster res “Majority Node Set” /priv MNSFileShare=\EDFS03\MNS_FSQ_E2K7CLUSTER
```

**Note**
Make sure to replace server name so it matches the name of the Hub Transport Server in your environment.

You will get a warning that all properties were stored but not all changes will take effect until the next time the resource is brought online, just like it’s shown in **Figure 32** below.
In order to force all changes to take effect, we will move the cluster group from one node to the other (will take the cluster group offline and online again). Do this using the below command:

**Cluster Group “Cluster Group” /Move**

When you have done so you will see that the cluster group is now online on **E2K7Node2**, as is the case in **Figure 33** below.
Now let’s verify the /Priv property is set correctly, this can be done by issuing the below command:

**Cluster Res “Majority Node Set” /Priv**

As you can see in Figure 34 below, this property has been set correctly for the purpose of this article series.
Configuring the Transport Dumpster

When using a CCR in your environment you should consider to configure the transport dumpster on the Hub Transport Server. Microsoft recommends that you configure the MaxDumpsterSizePerStorageGroup parameter, which specifies the maximum size of the transport dumpster queue for each storage group, to a size that is 1.25 times the size of the maximum message that can be sent. For example, if the maximum size for messages is 10 megabytes (MB), you should configure the MaxDumpsterSizePerStorageGroup parameter with a value of 12.5 MB. In addition they recommend you configure the MaxDumpsterTime parameter, which specifies how long an e-mail message should remain in the transport dumpster queue, to a value of 07.00:00:00, which is 7 days. This amount of time is sufficient to allow for an extended outage to occur without loss of e-mail. When using the transport dumpster feature, additional disk space is needed on the Hub Transport server to host the transport dumpster queues. The amount of storage space required is roughly equal to the value of MaxDumpsterSizePerStorageGroup multiplied by the number of storage groups.

You use the Set-TransportConfig CMDlet to enable and configure the Transport Dumpster. So in order to, for example, configure the maximum size of the dumpster per storage group to 25 MB with a dumpster life of 10 days, you would need to run the following command:

```
Set-TransportConfig -MaxDumpsterSizePerStorageGroup 25MB -MaxDumpsterTime 10.00:00:00
```

In order to see the MaxDumpsterSizePerStorageGroup and MaxDumpsterTime configuration settings, you can type Get-TransportConfig as I did in the figure below.

![Figure 35: Transport Configuration Settings](image)

Installing the Active Clustered Mailbox Role on E2K7Node1

It’s time to install the Exchange Server 2007 Beta 2 bits on each node, we’ll start with E2K7Node1. First, if you haven’t already done so, I recommend you copy the Exchange Server
2007 Beta 2 binaries to a drive locally on each node. When you have done so double-click Setup.com.

**Figure 36: Launching Exchange Setup**

The Exchange Server 2007 Installation Wizard will start, and as you can see **Step 1: Install .NET Framework 2.0** and **Step 2: Install Microsoft Management Console (MMC)** have already been completed.

**Note**

If you have installed Windows Server 2003 with Service Pack 1 on each node, you need to download Microsoft Management Console (MMC) 3.0 and install it manually (by following the link in **Step 2**). But since I’m using Windows 2003 R2 Servers in my test environment, the MMC 3.0 is installed by default.
As you can see we still need to complete **Step 3: Install Microsoft Command Shell (MSH)**, before we can start installing Exchange. Therefore click the link to download MSH then unzip and install it.
The Exchange Server 2007 Installation Wizard should refresh automatically, so now click Install Microsoft Exchange. Click Next then accept the License Agreement and Next once again. Decide whether you want to enable Error Reporting or not (a good idea to enable this functionality since the Exchange Product Group will receive any obscure errors you should experience in your cluster setup) then click Next.
Now select **Custom Exchange Server Installation** then click **Next**.
Figure 40: Selecting a custom Exchange Server installation

Tick Active Clustered Mailbox Role and click Next.
Figure 41: Selecting to install an Active Clustered Mailbox Role

Now select **Cluster Continuous Replication** then specify a name for the mailbox server (the name you want your Outlook clients to connect to) and a unique IP address on your public network. Finally specify the path for the clustered mailbox server database files or use the defaults (as is the case in this article series) then click **Next**.
Let the readiness check complete, and if no issues are found click **Next** to begin the installation.
The Exchange Server 2007 installation wizard will now copy the needed Exchange files, install and configure the Mailbox Role then finally create and configure the clustered mailbox server resources locally and create the object in Active Directory. When each step has been completed untick Exit Setup and open Exchange System Manager (yes this will be corrected in a later build), then click Finish. We don’t want to open the Exchange Management Console just yet, we’ll install Exchange on the second node first.

Installing the Passive Clustered Mailbox Role on E2K7Node2

Log on to E2K7Node2 with a domain admin account and do the exact same steps as we did when installing Exchange Server 2007 on E2K7Node1. Only difference is you should tick Passive Clustered Mailbox Role instead of Active Clustered Mailbox Role as shown in Figure 44 below.
Verifying the functionality of the Cluster Continuous Replication based Mailbox Server

It’s time to verify that our Exchange 2007 clustered mailbox server is working as expected. Let’s first open the Cluster Administrator and check whether the respective Exchange Resources have been created. If you take a look at Figure 45 below it looks good, we have both nodes listed in the left pane and all Exchange resources have been created and are currently owned by E2K7Node2.
Try to open the Exchange Management Shell by clicking Start > All Programs > Microsoft Exchange Server 2007 > Exchange Management Shell on one of the nodes, then type `Get-ClusteredMailboxServerStatus -Identity E2K7CCR`. As you can see in Figure 46 below the status of the clustered mailbox server is Online, and *E2K7Node2* is currently the active node.

Now that we have verified the clustered mailbox server is online, let's try to move the Exchange resources from node one to node two using the `Move-ClusteredMailboxServer` CMDlet. In the test environment used in this article, we do so by issuing below CMDlet:

```
Move-ClusteredMailboxServer -Identity:E2K7CCR -TargetMachine:E2K7Node1 -MoveComment:"This is a test!"
```

You're then asked to confirm this action, type Yes then hit Enter. After a while the clustered mailbox resources will have been moved to the first node.
Even though it’s possible to move the cluster resource groups between nodes using the Cluster Administrator console, you should always do so using the **Move-ClusteredMailboxServer** CMDlet as the **Move Group** task in the Cluster Administrator console isn’t Exchange 2007 aware.

Let’s also take a look at the clustered mailbox server in the Exchange Management Console. To do so click **Start > All Programs > Microsoft Exchange Server 2007 > Exchange Management Console**, then drill down do **Server Configuration > Mailbox**. Notice the clustered mailbox server which we named **E2K7CCR** is listed in the **Result pane** and that it’s recognized as a cluster server.

Let’s try to take a look at the transaction log file replay from one node to the other. The easiest way to do that is to generate a few log files by sending a couple of test messages with an attachment or two.
Note
Since the new transaction log file size in Exchange Server 2007 is 1 MB instead of 5 MB as was the case in previous versions of Exchange. It’s not required to attached files larger than 1 MB in order to generate these log files.

As you can see in Figure 49 below, the log files were replayed to E2K7Node2 within the same minute as they were generated on E2K7Node1.

Figure 49: Log file replay

Simulating a failover from E2K7Node1 to E2K7Node2

Okay let’s try to simulate a fail over from E2K7Node1 (which currently is the active node) to E2K7Node2, so that we can see what will happen from the Outlook client perspective. In order to switch from one node to the other we’ll issue below CMDlet which we also used earlier on in this article:

```
Move-ClusteredMailboxServer -Identity:E2K7CCR -TargetMachine:E2K7Node2 -MoveComment:"This is a test!"
```

When a manual move or a failover occurs, the balloon shown in Figure 50 will appear as all services needs to be stopped on E2K7Node1 before they are brought online on E2K7Node2.
Depending on the amount as well as size of the databases in your Cluster Continuous Replication setup, this will take somewhere between 10 seconds to approximately 1 minute, which shouldn’t cause panic for the end-users. When E2K7Node2 has taken over, the end-users will be notified that the connection to the Exchange Server has been restored (Figure 51).
Figure 51: Connection to the Exchange Server has been restored

Conclusion

You benefit from several advantages when you choose to install the Exchange 2007 Mailbox Server role in a Cluster Continuous Replication setup in your organization. The primary reason here is that you no longer have a single point of failure, when talking about the Mailbox/Public Folder databases. Should the database on one node crash, an automatic fail over to the other node containing the secondary database is completed. This also means you no longer need to use a shared storage system in the CCR setup, as is the case with Exchange 2007 Single Copy Clusters as well as cluster setup in previous versions of Exchange. In addition the two nodes in the CCR setup can even be placed in two different locations, as long as they belong to the same subnet. Not only that, the installation of the Exchange 2007 cluster has also been further simplified over previous versions. Since CCR setup makes use of log file shipping and replay to a secondary database, you also don’t have to do full online backups far as often as was the case in Exchange 200x and earlier versions. Last but certainly not least, the fail over process been improved in several areas now that the new file share witness model have been introduced.