Exchange 2007 Standby Continuous Replication on Windows Server 2008

Requirements

- Exchange 2007 SP1, Standard or Enterprise.
- Windows Server 2003 SP2 or Windows Server 2008 SP1, OS should match server, must be 64 bit version.
- All servers in the set need to have matching OS and Exchange versions.
- The StandbyMachines need their storage configured so that the logs and database files can be written to the exact same local path.

Benefits

- Site Resiliency
- Time Lag (which may be adjusted) for commitment of logs on remote copy, for recovery when the primary database is corrupted.
- Does not require the more expensive Enterprise Licenses.
- One Storage Group can have multiple targets, up to four, the source storage group can be clustered.
- If the Target is *not* clustered, it may host active Storage Groups, and or hold Transport and or CAS roles.

Disadvantages

- Does not provide for a smooth failover of resources, and therefore is not a High Availability Solution.
- Management only through PowerShell.
- Backups cannot be performed against the Standby Copy while SCR is active.

Implementing SCR

My Test environment had a private DNS space of brain.demo, a public DNS space of brainbuz.info, and a NETBIOS Domain name of BRAINDEMO. The Domain Controller was DC1, two Exchange Servers were EX1 and EX2.

For the most part my examples will use PowerShell code for all steps. In a production environment you will want to script your recoveries.

All examples and discussion refers to Windows Server 2008 x64, x32 is not supported in production (although there should be no material differences in it), and I have not repeated my lab with 2003.

Setting Up

Disable Circular Logging

While it is possible to use SCR while Circular Logging is enabled, SCR won't seed with fewer than 50 logs. Circular Logging overwrites log files, so even if you can

Two Annoying Things to Remember About PowerShell

SCR is implemented and managed entirely from within PowerShell. There are two Security Impairments (Microsoft considers them a feature, I consider them otherwise) in PowerShell that you need to be aware of.

ExecutionPolicy. ExecutionPolicy prevents PowerShell from executing scripts. Get-ExecutionPolicy will tell you what the current restriction is. **Set-ExecutionPolicy -executionPolicy Unrestricted** will turn it off. At least many PowerShell Starter Guides warn about this nuisance.

Reduction in User Privilege. On Server 2008, if you run PowerShell from a privileged account, PowerShell will reduce your privilege level. I wasted half a day with commands not working properly before I found the answer (The errors returned were meaningless or unrelated. Even if you consider this as a feature, not giving relevant errors is a bug!). This impairment which isn't well advertised is resolved by starting PowerShell with right-click, RunAs Administrator, even when you are logged in as the (root) Administrator. I have not yet found a way of turning it off. This impairment was not present on Server 2003.

And one Useful Thing

-WhatIf. This option supported by Exchange Management Commands will allow you to preview your command. It will at minimum tell you if you've typed a command that would execute. If you're lucky it may even provide useful feedback as to whether it will do what you intended to do.

get it to work, it isn't a good idea in terms of recovery. In a test environment this means it may be necessary to run an occasional backup solely for the purpose of truncating logs.

Confirm Your Storage Locations

Commandlets: **Get-StorageGroup** and **Get-Mailboxdatabase**

Add *–identity <name\SG>* to only obtain information about a specific storage group, or omit it for information for the exchange server where the command is being run. Omit the–property restriction for format-list to see a lot of information.

```
Get-StorageGroup -identity "ex1\primarystorage" `
      | format-list `
      -property `
      Identity, Name, LogFolderPath, SystemFolderPath,`
      CircularLoggingEnabled, StandbyMachines, DistinguishedName
```

Get-Mailboxdatabase (to learn the identity value for the next command)

```
Get-Mailboxdatabase -identity "ex1\primarystorage\mailbox database"`
      | format-list -property identity, DistinguishedName, `
      storagegroup, name, EdbFilePath
```

The standby copy of your database will have the exact same local path as the active copy, which means that you will need to rename / relocate some items from their default location. I also recommend choosing names without spaces so that you can omit the quotes when typing PowerShell commands.

The Commands to make the changes are Move-StorageGroupPath and Move-DatabasePath, -identity is required. Note that the database is specified in the move-database command, whereas the first command specifies the storage group.

```
Move-StorageGroupPath -identity "ex2\SecondStorage" `
      -LogFolderPath:"D:\LOGS\SecondSG" `
      -SystemFolderPath "D:\SystemFiles\SecondSG "

Move-DatabasePath -identity "ex2\SecondStorage\Mailbox Database" `
      -EdbFilePath "D:\Mailboxes\SecondStorage\secondstorage.edb"
```

Enable SCR

Enable-StorageGroupCopy requires the identity and the target (StandByMachine) server. In addition I set two extra parameters. **ReplayLagTime**, which controls the delay in committing logs to the SCR target. I've specified an hour. **TruncationLagTime**, which I've set to 10 minutes, is how long SCR waits to truncate logs after it commits them. There is no parameter to override the default 50 log lag and SCR waits until both conditions are met to commit a log. The date format for Lag Time is Days (up to 7) . Hours : Minutes : Seconds.

```
enable-storagegroupcopy -identity ex1\PrimaryStorage -replaylagtime
0.1:0:0 -standbymachine 'ex2.brain.demo' -truncationlagtime 0.0:10:0
```

Produces a warning.

```
WARNING: PrimaryStorage copy is enabled but seeding is needed for the
copy. storage group copy is temporaily suspended. Please resume the copy
after seeding the database.
```

Seeding the Copy

I do not know why the Enable-StorageGroupCopy does not support an option to "SeedNow", when it does have "SeedingPostponed". Many of the SCR commands were extended from LCR Commands (and you will see many LCR options in the full descriptions of them), in this case it appears that the default is for LCR to Seed.

Assuming you're doing this in a test or pre-production environment, replication won't happen until there are at least 50 logs. I used the Exchange 2007 Load Generator, which can be downloaded from Technet, to generate activity.

FYI, the Resume-StorageGroupCopy command must be run on the target.

```
Suspend-StorageGroupCopy `
      -identity ex1\PrimaryStorage -StandbyMachine ex2

Update-StorageGroupCopy `  (copies the EDB file)
      -identity ex1\PrimaryStorage -StandbyMachine ex2

Resume-StorageGroupCopy `  (copies log files and resumes)
      -identity ex1\PrimaryStorage -StandbyMachine ex2
```

The System Path .CHK File

The System Path location will contain a file with a name like E01.CHK, if you compare to the log files you will see that the logs start with the same first three characters. E01 is the default sequence on a newly created store. When you're trying to seed a replica, this file will not appear until seeding is completed.

Testing SCR Recovery

There are two methods that can be used to recover an SCR protected Database, Database Portability and Setup. Neither of these procedures are painless and quick.

This section deals with non-clustered servers, the final section deals with clustering.

Using Setup to Recover the Database

Setup /M:RecoverServer will move the exchange configuration from the old server to the new. This process is exactly the same as if you had been able to copy or restore the log, database, and checkpoint file from a failed server. This process is not appropriate in circumstances where you expect to be able to bring the original server back online. If you want instructions for this procedure I recommend Henrik Walther's Configuring Exchange Server 2007 book, the chapter can be read online if you search for setup /m:recoverserver at msexchange.org; also technet articles aa998656 and bb123496.

Database Portability

By creating an empty database, dismounting it and marking it as over-writeable by a Backup, we can replace its files with those of our SCR copy, and subsequently point the mailboxes from the original storage group to the new one.

First skip ahead a few pages and run Test-ReplicationHealth and Get-StorageGroupCopyStatus to make sure that you have healthy SCR replication. Remember that you may need to simulate some activity to reach the 50 log minimum. You can also look at the destination files to confirm that they are current within the lag parameters you specified.

Create an Empty Database for the StorageGroup

```
new-StorageGroup -Server 'EX2' -Name 'SG1' -LogFolderPath 'D:\LOGS\SG1' -SystemFolderPath 'D:\SystemPath\SG1'
```

```
new-mailboxdatabase -StorageGroup 'EX2\SG1' -Name 'MBX' -EdbFilePath 'D:\Mailboxes\SG1\MBX.edb'
```

Then mount it, dismount it, and finally configure it to accept a restore.

```
mount-database "ex2\SG1\MBX"
```

```
dismount-database "ex2\SG1\MBX"
```

```
Set-Mailboxdatabase EX2\SG1\MBX -AllowFileRestore:$true
```

Prepare the SCR local Copy for use. Run on the Recovery Machine. If the SCR Source is unavailable you will need to use an additional switch **–FORCE**.

```
restore-storagegroupcopy "ex1\primarystorage" -standbymachine ex2
```

The Code below shows how to re-point the empty Database and Storage Group you created to the location of the SCR files, but it works equally well to just move and rename those directories/files to those of your empty Storage Group. Without the ConfigurationOnly switch you would create a new empty database at the destination instead of just pointing to the files that were already there.

```
move-storagegrouppath "EX2\SG1" -systemfolderpath "D:\SystemPath\PRIMARYSG" -logfolderpath "D:\LOGS\PrimarySG" -configurationonly
```

```
move-databasepath "EX2\SG1\MBX" -edbfilepath
"D:\MAILBOXES\PRIMARYSG\PrimarySG.edb" -configurationonly
```

Either in the GUI or with this command, Allow the Database to be overwritten by a restore.

```
Set-Mailboxdatabase EX2\SG1\MBX -AllowFileRestore:$true
```

In theory you should be able to just mount the Database now!

```
mount-database "ex2\SG1\MBX"
```

Even if there was no loss in failure, the soft repair option of eseutil needs to be run before mounting, to commit all of the log files to the database.

Before running ESEUTIL copy the most recent .CHK file from the SystemPath of the Storage Group to the same directory as the database. Every time Exchange tries to mount the database it clears the AllowFileRestore flag, so prior to every mount attempt you must reset it to true.

```
Copy-Item "<%systempath%>\Xxx.CHK" "<%EDB File Location%>"
```

```
Set-Mailboxdatabase EX2\SG1\MBX -AllowFileRestore:$true
```

`eseutil /r E01 /LD:\LOGS\PrimarySG`	Try mounting if this succeeds.
`eseutil /p PrimarySG.edb`	Hard Recovery can take a *LONG* time.
`eseutil /r E01 /LD:\LOGS\PrimarySG`	Run it again after hard recovery.

```
mount-database "ex2\SG1\MBX"
```

At this point I've always gotten the database to mount. Once in testing I made an error and mounted the empty database and lost the contents of a test mailbox I logged into!

At this point you have the database in place, but all of your mailboxes are still pointing to the wrong server. You need to move the mailbox configuration only. We also don't want to move the Database specific system mailboxes.

```
Get-Mailbox -Database "ex1\primarystorage\mailboxdatabase" |where
{$_.ObjectClass -NotMatch
'(SystemAttendantMailbox|ExOleDbSystemMailbox)'}| Move-Mailbox -
ConfigurationOnly -TargetDatabase ex2\SG1\MBX
```

Reverting the Database to the Original Server.

To revert, reverse the procedure. Remove the original Storage Group from Exchange and delete all of its files. Enable SCR from the new Storage Group to the old server. Then follow this whole process over again. Another option is to simply move the mailboxes back; with a large database, watch transaction log space during the move.

Monitoring SCR

PowerShell Commands

Test-ReplicationHealth

```
Server          Check                       Result     Error
------          -----                       ------     -----
EX2             ReplayService               Passed
EX2             SGCopySuspended             Passed
EX2             SGCopyFailed                Passed
EX2             SGInitializing              Passed
EX2             SGCopyQueueLength           Passed
EX2             SGReplayQueueLength         Passed
EX2             SGStandbyReplayLag          Passed
```

Get-StorageGroupCopyStatus -Identity ex1\primarystorage –StandbyMachine ex2 | format-list

Test-ReplicationHealth has no parameters and will only report the status for the local machine, Get-StorageGroupCopyStatus requires identifying parameters and can be run from any Exchange Management Station.

Reliability and Performance Monitor

Perfmon data relevant to SCR is captured under MSExchange Replica Seeder and MSExchange Replication, here is a sample output of the report format.

\\EX2	
MSExchange Replica Seeder	**sg1**
Seeding Finished %	100.000
MSExchange Replication	**sg1**
Copy Queue Exceeds Mount Threshold (CCR Only)	0.000
CopyGenerationNumber	178.000
CopyNotificationGenerationNumber	178.000
CopyQueueLength	0.000
Failed	0.000
Initializing	0.000
InspectorGenerationNumber	178.000
ReplayBatchSize	0.000
ReplayGenerationNumber	128.000
ReplayGenerationsComplete	0.000
ReplayGenerationsPerMinute	256.000
ReplayGenerationsRemaining	0.000
ReplayNotificationGenerationNumber	178.000
ReplayQueueLength	50.000
Suspended	0.000
TruncatedGenerationNumber	0.000

Troubleshooting

When I was working on this document I came across this error several times (it correlates to MSEXCHANGEREPL 2070 in the EventLog).

Test-ReplicationHealth

```
Server          Check                       Result     Error
------          -----                       ------     -----
EX2             SGCopyFailed                *FAILED*
      Standby Continuous Replication for storage group
      'EX1\PrimaryStorage' is in a 'Failed' state on server 'EX2'. The
      error message is: The Microsoft Exchange Replication Service
      encountered an error while inspecting the logs and
      database for EX1\Primary Storage on startup. The specific error code
      returned is: Microsoft.Exchange.Cluster.Replay.FileCheck
      LogfileMissingException: File check failed : Logfile
      'D:\Logs\PrimarySG\E0000000001.log' was not found…
```

This error could be caused by a lingering complication of Circular Logging, but was probably caused by leaving my lab servers turned off for too long.

The solution is to **Suspend-StorageGroupCopy**, and then reseed.

Disabling SCR

```
Disable-StorageGroupCopy `
      -identity ex1\PrimaryStorage -StandbyMachine ex2
```

SCR in a Clustered Environment

Disclaimer: I have only experimented with Continuous Cluster Replication configurations, but Single Copy Cluster configurations should work in a similar manner.

Standalone Target

This SCR configuration uses a Standalone Target rather than another cluster. In this configuration recovery is exactly the same as if the Exchange Cluster Source and the Destination were both regular un-clustered Exchange Servers.

It should be possible to replicate a clustered server's database to a Standalone Server running the Standard Edition of Exchange. I have not tested this with valid licenses. The versions are determined by which license key is input and the feature is supported fully in both versions, so it should work, unless Exchange is too smart and checks the version too carefully.

The drawback of Standalone Target is that you are recovering the mailboxes to an un-clustered machine. To protect those mailboxes with a cluster, they then have to be moved to another server. With the Setup Recover Clustered Mail Server method, you only have to bring a second node online within the cluster.

Setup Recover Clustered Mail Server

This method moves the exchange cluster to the Standby Server. The Standby Server has to be built as a Passive Exchange 2007 Mailbox Server. If called on, the Standby Server replaces the original servers as members of the cluster. This process can be used for migrating a server to new hardware; the cluster name will be preserved.

Setting TTL for the Exchange Cluster Network Name.

If a cluster needs to be moved to the Standby Machine, and cannot take it's IP address with it, clients and other exchange servers will not be able to reach the new address until cached copies of the DNS A record expire. Microsoft recommends a 5 minute TTL for this record.

Cluster.exe res <exchange resource name> /priv HostRecordTTL=300

Building a Standby Passive Node.

Prepare a single node as if it were going to be added to your CCR cluster. It will need all Exchange Pre-requisites and the failover clustering feature.

Create the Cluster:

```
CLUSTER.EXE /CLUSTER:<newclus> /CREATE /NODE:<nodename>
                        /IPADDRESS:"<ip of cluster>/<subnet mask>"
```

Using the GUI, install the Passive Mailbox Role. I recommend using the GUI over the command line for this one operation since it allows you to explicitly select the Passive Mailbox Role.

Reboot your server.

Configure your server as the Standby machine as previously.

Recovering the Cluster to the StandbyMachine

On the Standby Machine, recover the local copy with the Recover-StorageGroupCopy commandlet (see earlier section). In an orderly transition you would dismount the databases and then use the Stop-ClusteredMailboxServer command on the old server.

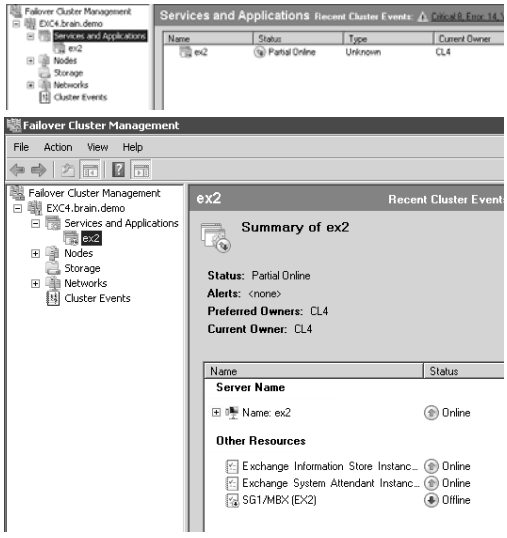
IN DNS, update the A record for the Exchange Virtual Server to the new IP address.

Run **ipconfig /flushdns** to clear the names cache on the Standby Machine.

Recovering the Cluster must be performed at the command line.

Setup /RecoverCMS /CMSName:<clusname> /CMSIPAddress<new ip>

Failing to mount the database will make setup report failure. Check the Cluster GUI, if the resources have been created, you're successful.



Set the database to allow over-write by restore. Run eseutil /r and try mounting the database. If it still fails to mount run a hard recovery (/p), set the restore flag, and try to mount again.

Repairing the Original Servers

When you are able to bring the original servers back online, you will need to execute the command:

Setup /ClearLocalCMS /CMSName:<servername>

When it completes you can configure your original servers as StandbyMachines and reverse the process to bring the cluster back home.

References:

Technet Articles:
bb738150, bb738132, bb690985, aa998656, bb123496, bb123479

Technet Exchange Online Documentation

My own field experience and test environment studies