VERITAS®

VERITAS WHITE PAPER

Using VERITAS Storage Replicator™

for Windows NT and Windows 2000

Replicating Microsoft Exchange 5.5 and Exchange 2000 Databases

4 April 2002

VERITAS®

Table of Contents

Introduction

Understanding VERITAS Storage Replicator™ and Microsoft Exchange

How the Two Systems Work

VERITAS Storage Replicator™ 2.1 *for Windows NT and Windows 2000* is designed to continuously monitor the state of data on a VERITAS Storage Replicator system and record changes made to data on the system's disks. Those changes are selectively mirrored to another system (the Target server) via a network connection. There is no requirement for physical proximity between the Source and Target servers. Storage Replicator allows users to maintain a mirror copy of their data so if the Source server fails the Target server will have a complete and consistent record of all changes made to the Source, which can be retrieved with minimal downtime.

Many customers have expressed the desire to use this powerful capability in their Microsoft Exchange environments. To support that need, best practices have been developed for configuring Storage Replicator to replicate Exchange for migration and disaster recovery. There are some special requirements for using Storage Replicator with Microsoft Exchange because of the way the Exchange storage subsystem works. This paper describes how to configure Exchange 5.5 or Exchange 2000 and Storage Replicator for successful migration of Exchange services and continuous replication for disaster recovery from a Storage Replicator Source server to an alternate Target server.

How Exchange Stores Data

Understanding how Exchange stores data is critical to understanding how to configure and operate Storage Replicator with Exchange 5.5 or Exchange 2000.

The Database System

Exchange 5.5 stores all its directory and message data in a trio of databases: *priv.edb* holds mailbox data, *pub.edb* holds public folder data, and *dir.edb* holds the server's copy of the organizational directory. These are the files we will be replicating (see below). These files look like ordinary files, but they are actually databases maintained and accessed by Exchange's database subsystem, the Extensible Storage Engine (ESE). ESE provides an engine for indexing, storing and retrieving data in 4 KB chunks called pages; Exchange uses the ESE I/O routines to organize those pages into mailboxes, messages and so forth.

Several changes to the database organization were introduced with Exchange 2000. First, the organizational directory is no longer maintained by Exchange 2000 itself. Rather, it is integrated with the Windows 2000 Active Directory and stored on a Windows 2000 domain controller. Since the Microsoft Active Directory provides its own replication tools, it is not necessary to configure Storage Replicator to replicate this information in an Exchange 2000 environment. All references to the DS in this paper apply only to Exchange 5.5.

Additionally, Exchange 2000 adds the capability of supporting multiple private and public *information stores*, each contained in its own database. These databases can be grouped into *storage groups*. All databases within an Exchange 2000 storage group share the same set of log files. The ability to segment information stores in this way helps to provide speedier recovery in the event of corruption. If only one database is corrupted, the others are not affected, and some users may still be able to send and receive mail.

Finally, to support multimedia attachments in their native format, Exchange 2000 splits each information store into 2 pieces – an *.edb* file containing rich text information, and an *.stm* file containing native multimedia (MIME) formatted data. When using Storage Replicator to replicate Exchange 2000 data, it is important to replicate both the .edb and the .stm files, as well as their storage group's associated log files.

ESE is a transactional database. Each change or set of changes to the database is recorded as an individual transaction; these transactions are stored in transaction log files before they are committed to the database. If a database is damaged or corrupted, the transactions can be replayed from the log files. This can often restore a database to its original pristine state without requiring a full restore from a backup. In order for Storage Replicator to be useful as a data protection tool, it is necessary that changes to both the databases and the transaction logs be copied to the Target system.

Log and Checkpoint Files

Although Exchange treats the log as a single entity, it is actually a set of files, each exactly 5 MB (5,242,880 bytes), even if there aren't any transactions in it. The Exchange Information Store (IS) and Exchange 5.5 Directory Service (DS) each maintain their own log files (mdbdata\edb.log and dsadata\edb.log, respectively). As transactions for each service occur, they are written to the appropriate log file. When the log file fills up, the DS or IS service renames it, using a sequential hexadecimal ID (the first file is edb00001.log, the second is edb00002.log, and so on). These renamed log files are called generations; edb.log represents the highest, or most recent, generation. Note that just because a log file is full doesn't mean its transactions have been committed — all commitments happen according to the rules outlined in "The Logging Process," below. The log files contain a number of useful items that are used if the logs have to be played back during server recovery, including the full path to the database files, information about which generation of log data is in the file, a signature and time stamp for the data, and a signature for the database. This header information enables the store to make sure that each log file is replayed into the correct database file, and to balk if you do something like try to restore files from one machine to another with a different name.

The log files also contain information about the transactions themselves. For each transaction, the log records the type of transaction (e.g., whether the transaction represents a change, a rollback of a previous change or a commit of a previous change). These transactions actually record the low-level modifications to individual pages and tables within the database. When the DS and IS services are shut down normally, any transactions that have been made to the in-memory copy of the database are committed to the disk version, and the checkpoint file is updated to reflect which transactions have been committed. If the service is shut down abnormally (say, by a power failure), when it restarts it will scan its inventory of log files and play back any uncommitted transactions from the log files to the database. This means that it's very important not to move, delete, edit or otherwise disturb the log files until their transactions have been committed. How do the services know which transactions have been logged? The IS and DS services maintain separate checkpoint files (dsadata\edb.chk and mdbdata\edb.chk). Whenever a transaction is committed, the checkpoint file is updated to point to that transaction. The services use the checkpoint file at startup time; if this file is present, transactions are played back from the checkpoint to the end of the last available log file. The checkpoint files tell the store which transaction log files contain uncommitted transactions — those are the files that would be needed in case of a crash. If the checkpoint file is missing or damaged, Exchange can scan each log file and check whether its transactions have been committed, but this is much slower than using the checkpoint files.

The Logging Process

Logging transactions is a good way to keep the database consistent. However, performance costs may be involved. A simplistic logging mechanism would just log transactions to a file, and then periodically inject them into the database. The Exchange logging process is quite a bit smarter; it works like this:

1. A change (such as the arrival of a new message or the deletion of a directory object) occurs.
2. A new database transaction is created by the directory or IS services. The transaction only reflects data that's changed:

   For example, if you open a draft message in your mailbox, edit it and resave it, the transaction will contain only your changes, not the entire draft.
3. The time stamp on the page affected by the new transaction is updated.
4. The transaction is logged to the current generation of log file for the service that owns it. Transactions are written to the log file in sequence, meaning that writing to the log file is fast — all writes occur in sequence, with no randomaccess seeking. Once the transaction has been logged, Exchange assumes that it will be properly registered in the database and goes about its business.

5. The transaction is applied to the version of the store database cached in memory. The store never records a transaction to the cached database until after the transaction has been logged.
6. When the log file hits its maximum size, the service that owns it renames it and creates a new log generation. This log file will stay on disk until it is purged during an online backup.
7. When load permits, Exchange copies the transactions from the cached copy in memory back to the disk version of the database. This so-called lazy commit strategy means that at any point the complete, consistent database actually is composed of data from the database file on disk, data from the database copy in memory and as-yet-uncommitted transactions.

   When the DS or IS services are shut down normally, they attempt to commit any outstanding transactions from the in-memory database, but not from the log files. If the service shuts down abnormally, and the transaction files remain intact, when the services restart they will replay transactions starting at the checkpoint. If the transaction files are missing or partially damaged, the DS and IS services will do the best they can to commit any transactions that can be recovered. However, in such cases the databases may not be mountable until they are repaired.

Log Files and Backups

Exchange comes with a modified version of the standard Windows NT backup application, which can back up the DS and IS databases without stopping the Exchange services. Full and incremental backups will purge log files whose transactions have all been committed — both of these backup types record all changes to the IS and DS databases, so the log files are no longer needed. Differential backups require a complete set of all log files since the last full backup, so differential backups don't purge any files. If you enable circular logging (in which the same log file is overwritten with new logs), you won't be able to do incremental or differential backups. (If you stop the Exchange services and do an offline backup, you can do incremental and differential backups, but you lose the Exchange-aware features of ntbackup.)

One key feature of Exchange-integrated backup products (including VERITAS NetBackup™ and VERITAS Backup Exec™) is the way incoming transactions are handled during backup. Here's how the process works for a full backup:

1. The backup starts. Any transactions that are in the transaction log but haven't been committed to the database are committed. The various checkpoint (.chk) files are updated to reflect which transactions have been committed and which are still outstanding.
2. The backup proceeds. Each database file is backed up in turn, in 64 KB chunks.
3. If new transactions arrive for a database that's currently being backed up, they are stored in two places: the transaction logs for that database and the corresponding patch file (priv.pat, pub.pat, and dir.pat — one for each database). The patch files hold copies of incoming transactions that would ordinarily apply to database pages that have already been backed up.
4. When the database file is completely backed up, the patch files insert the new transactions into the database. While these files are being backed up, new transactions are stored in the log files.
5. When the patch files are completely backed up, the remaining log files are backed up, too. Normally the last log file will be partially full. (Each log file is exactly 5 MB, but fewer transactions can be in the last file.) Incremental backups work differently: They copy just the log files, not the main databases. This five-step process allows Exchange to continue running while a backup is in progress. Incoming transactions generate log files; on an active system, there may be several log files generated during the backup. There's nothing wrong with this, because the backup process is designed to handle it. However, it can make the patch files grow pretty large, so they take extra time to back up (and restore).

How VERITAS Storage Replicator Works

VERITAS Storage Replicator™ provides real-time data protection for Windows NT or Windows 2000 environments by replicating critical files to one or more offsite servers. Replicated files on the Target server are constantly updated with the changes made to the original files on the Source server, so an exact copy of each file is always available. Storage Replicator

allows administrators to use a management console to select data on a Source system and have it replicated to a Target system. It is possible to select individual files, folders or volumes for replication. Storage Replicator installs a file system-level driver that watches write requests to the physical volumes, captures the data to be written, copies it to an outbound journal and then replicates it to the Target server. Normally, the Target server is on the same network as the Source; when it is time to migrate services from the Source to the Target, the necessary data will already be on the Target.

The Admin Console

The Storage Replicator Admin Console is an application that provides information about the replication configuration and replication processes. It communicates directly with the replication software on servers in your network. With the Admin Console, users can deploy and configure servers for replication. The Admin Console is used to create replication jobs; each job contains one or more rules that specify which files, folders and volumes should be included (or excluded) from replication. In addition, the Console lets you start, stop, pause and reset previously defined replication jobs.

The Replication Management Server

The Replication Management Server (RMS) holds configuration data for the replication system; it also starts and stops the replication process by initiating replication jobs according to user commands or a predetermined schedule. The RMS drives the replication process according to the configuration settings sent from the console. The RMS is also the repository for job logs, alerts and histories. (Server logs are maintained on the individual servers.)

The Replication Service Agent

With Replication Service Agent (RSA) software installed on a server, that server can be designated a Source server, a Target server or both. This software can be installed either manually at the server itself, or via the Console on another machine, "pushing" the software installation across your network.

The Command Line Interface (srTool)

srTool is a command line utility program that allows the administrator to create, configure and control replication jobs without using the Console. srTool incorporates a powerful command language that enables administrators to easily automate many complex administrative tasks.

Replication Planning for Microsoft Exchange

Approaches Using VERITAS Storage Replicator

Two basic approaches are discussed in this white paper. The first is what is called "Graceful Migration" and is used in planning for upgrades to existing servers, hardware upgrades, or migration of the Exchange software and database files to another server. The second approach is for disaster recovery in the event that the Exchange server becomes unavailable. Graceful Migration requires a one-time set of procedures for migrating the Exchange server and all of its associated files to another Target server with as little interruption as possible to Exchange users. Disaster recovery is designed to minimize the impact of a disaster on the Exchange server. In a disaster, the Exchange server may be totally unavailable and potentially unrecoverable. This solution provides for an alternate server to be available and up-to-date with the latest copy of the Exchange software installed and the current database and associated files available to bring online in the event of a disaster.

Using VERITAS Storage Replicator for Graceful Migration of Microsoft Exchange 5.5 or Exchange 2000

By using VERITAS Storage Replicator, the Microsoft Exchange Database, log files and associated files can be replicated to a secondary server as part of an upgrade strategy. Hardware can be replaced, or the primary Exchange server can be migrated to a new location. Many companies today have gone to a centralized model and use a corporate data center to house their most active servers. The basic steps involved in migrating the primary Exchange server (database and associated files) are as follows:

1. Installation of VERITAS Storage Replicator on the primary Exchange server and new (Target) server
2. Installation of Exchange on the primary server
3. Installation of Exchange on the new server
4. Creating a replication job using the Storage Replicator console
5. Running the replication job
6. Validating the new Exchange server
7. Shutting down the primary Exchange server
8. Adjusting the network configuration for the new Exchange server
9. Bringing the new Exchange server online

Considerations When Planning for Graceful Migration

Using Storage Replicator to migrate Exchange is a good way to ensure that Exchange can continue to run on a new server maintaining the same set of users and similar network server configuration. The process is fairly simple to set up and can be done while the existing Exchange server is running. Once the Exchange server has been migrated to the new (Target) server, the Exchange Database can be validated to ensure everything is in order before bringing the server online.

Using VERITAS Storage Replicator for Disaster Recovery of Microsoft Exchange 5.5 or Exchange 2000

By using Storage Replicator, the Microsoft Exchange Database, log files and associated files can be replicated to a secondary server. In the event that the Exchange server is unable to perform its normal functions, the secondary (Target) Exchange server can be used to replace the primary Exchange server. The basic steps involved to ensure that the Exchange Database and associated files are replicated correctly and available to recover are as follows:

1. Installation of VERITAS Storage Replicator on the primary Exchange server and secondary (Target) server
2. Installation of Exchange on the primary server
3. Installation of Exchange on the secondary server
4. Creating a replication job using the Storage Replicator console
5. Running the replication job
6. Making a copy of the secondary Exchange database
7. Validating the copy of the secondary Exchange database
8. Determining when it is necessary to bring up the secondary Exchange server to take the place of the primary Exchange server
9. Shutting down the primary Exchange server
10. Adjusting the network configuration for the secondary Exchange server
11. Bringing the secondary Exchange server online

Considerations When Planning for Disaster Recovery

Using VERITAS Storage Replicator to replicate Exchange is a good way to ensure that you can recover from a problem by having a secondary copy of Exchange on an alternate server. The process is fairly simple to set up, and the secondary server can be brought online in the event that the primary Exchange server fails, the OS becomes corrupt, or if there's a hardware failure. The risks involved in using this approach are related to data corruption. In the event of a database corruption, the secondary Exchange server may also be corrupted because Storage Replicator will replicate Exchange and all associated files exactly as they occur on the primary Exchange server. A method is described in Appendix C on how to ensure against data corruption using Storage Replicator and other VERITAS solutions.

Installation

Installing VERITAS Storage Replicator on the Primary and Secondary Exchange Servers

Before Exchange can be replicated, each server involved in the replication process must have VERITAS Storage Replicator installed. (See the *VERITAS Storage Replicator User Guide* for instructions on how to install Storage Replicator.) Once Storage Replicator has been installed and the servers rebooted, run the Administration Console to verify that the primary and secondary servers are available for replication.

Installing Exchange on the Primary and Secondary Servers

Now you will need to install Microsoft Exchange on both the primary and secondary servers. If Exchange is already in use by the primary Exchange server, then you will only need to install Exchange on the secondary Exchange server. Installation details differ somewhat between Exchange 5.5 and Exchange 2000, so this paper will describe two different methods of installing these products.

Exchange Virtual Servers

When users read email using a mail client such as Microsoft Outlook, Outlook Express, or web access, they configure their client software to point to a known Exchange Server name. In the event of a server migration or disaster, when a standby server takes over, it is important to provide the original server name for client access. When a server name moves from one physical server to another, we often refer to the name as a *virtual server* name. This paper describes two different methods of providing a virtual Exchange Server name. With Exchange 5.5, the registry on the standby machine is modified during Exchange installation to make Exchange 5.5 believe that it is being installed on the primary (original) server. After migration or disaster failover, the standby machine must be rebooted and actually takes on the name of the original primary machine. With Exchange 2000, we rely on the capabilities of Microsoft Cluster Server (MSCS) and the cluster aware features of Exchange 2000 to provide the virtual server name functionality. No reboot is required in this case.

Installing Exchange 5.5 on the Primary and Secondary Servers

The Exchange 5.5 Directory Service database file (dir.edb) is machine-specific. Since replication takes place between two different servers, the Target machine will need to be adjusted (via a registry key) so that the installed Exchange software will reference the correct Exchange database. Use the following steps to install Exchange 5.5:

1. Install Microsoft Exchange on the primary server. Make sure that it is installed on an NTFS-formatted drive. Note the directory paths where the database files and binaries go (and remember, they may change if you follow Microsoft's suggestion to run the Performance Optimizer after the product is installed). When the secondary Exchange server is installed, it will need to use the same directory path structure that was used by the primary Exchange server.
2. Install the latest Exchange 5.5 service pack (currently SP4) and reboot the server.
3. Once the installation is complete on the primary server, run the Performance Optimizer. It may suggest moving the database or log files; you can choose to accept or reject its suggestions. Note the actual paths for all database and log directories.
4. Switch to the secondary Exchange server and change its computer name in the registry by running REGEDIT and adjusting the following registry key:

   HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\ComputerName\ActiveComputerName

   Change the "ComputerName" value to the primary Exchange server name. That will be the name that the secondary Exchange server will be changed to when it is used to replace the primary Exchange server.

Note: Exchange should be installed on the primary server first and then installed separately on the secondary server to avoid conflicts. Don't try to install on both servers at once.

5. Install Exchange to an NTFS-formatted drive on the secondary server. The installation directory must be the same on both machines, and you must use the same account to install Exchange.
6. After installing Exchange on the secondary server, install the same Exchange service pack as on the primary server.
7. Once installation is complete, reboot the secondary server. Note that the Exchange services will not start because the computer name has now reverted back to the original name of the secondary server.
8. If this is a new Exchange installation and this is the first time that Exchange has been set up, you will need to create all user mailboxes and distribution lists on the primary Exchange server and verify that users can send and receive mail.

Installing Microsoft Cluster Server on the Primary and Secondary Servers for Exchange 2000

With Exchange 2000, we rely on the capabilities of Microsoft Cluster Server contained in Windows 2000 Advanced Server. For this replication solution, MSCS is only used to provide the virtual server name for Exchange 2000, as described in the previous section. For this solution, both the primary and secondary machines must have a minimum of 2 SCSI busses to allow installation of MSCS. The system drive is attached to one bus, while disks containing Exchange data are attached to the other bus (or busses).

For the Storage Replicator disaster recovery solution, the primary and secondary machines are not members of the same MSCS cluster. Rather, they are independent machines, and may actually be separated by a long distance and connected by a Wide Area Network (WAN). Each machine can be configured as a "one-node" MSCS cluster, which does not require any shared SCSI storage. It is also possible to have each machine be a member of its own two-node (or multi-node) MSCS cluster. The important feature that MSCS provides in this case is the virtual server name capability.

If you have not already installed Microsoft Cluster Server on both machines, use the following steps to start installation of MSCS to form two separate "one-node" clusters.

1. From the Windows Start Menu, select Programs | Administrative Tools | Configure Your Server
2. From the "Configure Your Server" window, select "Advanced | Microsoft Cluster Server" and follow directions to install MSCS. Select "Form a new Cluster" rather than "Join an existing Cluster"

   After MSCS is installed on both machines, you must configure an Exchange cluster *resource group* on each machine. Use Microsoft Cluster Administrator to perform the following steps:
3. On the primary server, configure an Exchange cluster resource group to contain all the *physical disk* resources for disks that will contain Exchange databases and log files.
4. Add an *IP Address* and *Network Name* resource to this group. This IP address and network name will be the *virtual server* address and name for your Exchange 2000 server. All mail clients will use this name and address when contacting the Exchange server.
5. Bring the Exchange cluster resource group online.
6. Install Exchange 2000 Server. Exchange 2000 installation should notice that it's running in a cluster and will offer to install the cluster version of Exchange 2000. Accept its offer.
7. Finish creating the Exchange cluster resource group by using Cluster Administrator to create a new resource of type "Exchange System Attendant". Make this resource a part of the Exchange cluster resource group. The Exchange cluster wizard will create several other cluster resources to control the operation of the Exchange 2000 virtual server.
8. Bring the Exchange cluster resource group online.

   At this point, you should have a running Exchange 2000 server. You can use Exchange Administrator to configure alternate directories for Exchange data and log files, if desired. Remember, all the data and log files must reside on disks that are members of the Exchange cluster resource group.

You can now configure client mailboxes using the Active Directory Users and Computers utility. Use the Exchange 2000 virtual server name as the name of the Exchange server when you configure the mailboxes. You should test the operation of Exchange 2000 to make sure it is functioning as expected.

Now it's time to configure a virtual server and install Exchange 2000 on the secondary machine. To avoid conflict with the primary machine during secondary installation, you must take the primary Exchange virtual server and IP address offline. Use the following steps:

9. Use Cluster Administrator on the primary machine to take the Exchange resource group's IP address offline. This will cause all Exchange services and the network name resource to go offline too.
10. On the secondary machine, repeat steps 3 through 8 to install and configure an identical Exchange 2000 virtual server.

    Note: The physical disk drive letters should be identical to those on the primary server. The IP address and network name resources should also be identical. This allows the secondary virtual server to come online and appear to have the same network identity as the primary server after migration or a disaster scenario.
11. Use Exchange Administrator on the secondary machine to make sure all Exchange database and log files are stored using the same paths as on the primary server. The data and log file path layout must be identical on both machines to enable migration or disaster failover.

    At this point, Exchange 2000 is active on the secondary machine. Of course the Exchange data and log files do not have the same content as the primary server yet.

    Finally, the Exchange virtual server on the secondary machine must be put offline, and the virtual server on the primary machine activated again.
12. Use Cluster Administrator on the secondary machine to put the Exchange resource group IP address offline. This wlil also put the Exchange services and network name offline. Remember to leave all the group's physical disk resources online so the Storage Replicator can copy data from the primary system's disks to the secondary's disks.
13. On the primary machine, use Cluster Administrator to bring the Exchange cluster resource group back online. The Exchange 2000 server is now available for production use.

Replicating the Primary Exchange Server

Creating a VERITAS Storage Replicator Job to Replicate the Primary Exchange Server

Now you will need to create a replication job using the Storage Replicator Console. From either system, run the Storage Replicator Admin Console. Use the following steps to create the replication job.

1. Open the Storage Replicator Console and create a replication job using the New Job Wizard.
2. Create a one-to-one (standard) replication job.
3. Give the job a name and description.
4. Specify the options for the job:
   - Prescan (yes)
   - No Changes on Target (yes)
   - Exact Replica on Target (yes)
   - Continue Replicating After Synchronization (only if you're setting up the replication job for disaster recovery as opposed to a migration strategy).
5. Specify the Source (primary Exchange) and Target (secondary Exchange) servers in the Storage Replicator server dialog. Use the physical names of the machines, not the Exchange virtual server name.
6. Specify the data to be replicated: In the case of Exchange 5.5, you will need to replicate the mdbdata and dsadata directories. Note that the Performance Optimizer may have located the individual databases and their logs on different drives; make sure you're replicating all of the logs and databases. In the case of Exchange 2000, it is best to replicate the entire EXCHSRVR directory for each disk containing Exchange data or log files.

   Storage Replicator will select a default path on the Target unless you manually override the Target path. You must configure the replication job so the source and target paths are identical. Create a set of rules with a custom Target path for the following: (in this example the NTFS Drive that Exchange was installed on happened to be the D: Drive) D:\exchsrvr\mdbdata and D:\exchsrvr\dsadata as well as E:\exchsrvr\mdbdata and E:\exchsrvr\dsadata to the secondary Exchange server with the same exact directory structure (path).
7. Run the new replication job.
8. Using the Storage Replicator Console, verify that the replication job has run correctly and that there were no errors.
9. Verify the replicated data. In the case of a one-time non-continuous replication job, replication will complete after synchronizing the source and target files. Once replication is complete, you may want to verify that the data is the same on the Target as on the Source. This is done by checking the directories on the Target server and making sure that the files on both the Source and Target servers are the same size. You can also use the validation steps discussed below to check the replicated data's logical integrity.

Using the Exchange Utilities to Validate the Exchange Database

One Time Replication for Migration

If you have configured the Storage Replicator job as a one-time job for Exchange migration, the replication job will eventually complete. Once replication is complete, you'll need to validate the Exchange databases on the Target to ensure that they contain complete and correct data. The quick-and-dirty way to tell if a database is useful is to attempt to mount it and start the Information Store or Directory Store services. If there are gross problems with the database, this will immediately make them evident. However, in some cases it's possible to mount a corrupted database, which may not be desirable.

The first, and simplest, test is to dump the database headers of the database files themselves. If the database header is corrupt or otherwise not intact, the database in question won't be mountable. To perform this check, use the eseutil /mh command. The /mh switch tells eseutil to check the database headers for integrity. If the database doesn't pass this test, it is not usable and will have to be repaired (see below) or re-replicated. Typical runtimes for this test are 15 seconds or less.

If the database headers are OK, the next step is to test the integrity of the entire database using eseutil's /g switch. This switch causes eseutil to verify the integrity of the database pages, tables and links; it checks for orphaned pages, missing links and other symptoms of database damage. This test may report errors even on a correctly replicated database, since it's checking the quality of data written to the database, not whether the data was written correctly. Storage Replicator will happily replicate database corruption that occurs on the Source — make sure you do periodic integrity checks on the Source server too. Typical run times for this test are 4 GB to 8 GB an hour. If eseutil finds errors, it may be possible to repair them using other eseutil switches. (See Appendix C for more details on repair operations.)

The third test is to use the isinteg command, which checks the logical integrity of the data in the database. Isinteg looks for things like orphaned attachments, bad single-instance pointers and other artifacts that result from bad message data being written correctly to the database. The most exhaustive test is to use the "test - alltests" flag, which tells isinteg to run all 20-plus logical tests in its arsenal:

```
isinteg –pri –priv.edb –test alltests
```

Continuous Replication for Disaster Recovery

If you have configured a continuous replication job for disaster recovery, you must stop the replication job before using the utilities described above. While continuous replication is running, the target files are marked as read-only, which may prevent some of the Exchange utilities from doing their job.

After you have validated the databases, you can restart the replication job. Storage Replicator will then resynchronize the files between the primary and secondary machines and start continuous replication again.

Changing Servers

To allow the standby server to take over the role of primary Exchange server, several steps must be taken. These steps differ depending on whether you are performing a graceful migration or using replication for disaster recovery. There are also differences between the two versions of Exchange Server software.

Exchange 5.5 — Shutting Down the Primary Exchange Server

If you are using replication to provide a graceful migration path to a new Exchange Server, you will have to shut down Exchange services on the primary server prior to finishing the migration. Of course, for disaster recovery, the primary server will already be inactive, so the steps shown in this section are unecessary in that case. Before you can bring the secondary Exchange 5.5 server online, you will need to shut down the primary Exchange server. This is because the Exchange database contains the name of the server on which it originated. Since we replicated the primary server's database, that database can be mounted only on a server with the same name. We're moving the database to a different server, so we must rename the secondary server so it has the same name as the primary machine. The first step in this process was mentioned earlier. (After installing Exchange, it is necessary to set the value of HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\ComputerName\ActiveComputerName on the secondary server to contain the name of the primary Exchange server.) In addition, the primary Exchange server must be removed from the list of servers found in the domain that the server is attached to. Do the following steps:

1. Shut down the primary Exchange server.
2. Remove the primary Exchange server's computer account from the domain.

Exchange 5.5 — Adjusting the Network and Adding the Secondary Server

Before you can bring the secondary Exchange 5.5 server online, you will need to change the name of the server, reboot the server and add it to the domain. Do the following steps:

1. Rename the secondary Exchange server to the name of the primary Exchange server. Note that this will require a reboot.
2. Use Server Manager to add the secondary server to the domain.
3. Start the Exchange System Attendant service. Verify that it started without error.
4. Start the Exchange Information Store service. Verify that it started without errors. If the IS won't start, check to make sure that per Microsoft Knowledge Base article Q248124, the correct server name is set in the registry keys (This Server, DSA Computer and MTA Computer) under HKLM\System\CurrentControlSet\Services\MSExchangeIS\ParametersSystem.
5. Start the Exchange directory service. Verify that it started without errors.

You should now be ready to run Exchange as the new primary Exchange server.

Exchange 2000 — Shutting down the Primary Exchange Virtual Server

With the MSCS virtual server configuration, it is much simpler to transition from primary to secondary server machine. As with the Exchange 5.5 description above, it is not necessary to shut down Exchange services on the primary machine in the case of disaster recovery. In that case, the primary machine is not running. But for graceful migration scenarios, you must first stop Exchange services on the original primary machine:

1. On the primary machine, use Cluster Administrator to take the Exchange resource group IP address offline. This will also stop the Exchange services and will take the group's network name resource offline.

Note: Do not put the entire Exchange resource group offline. The physical disk resources in the group must stay online so replication of files on those disks can finish.

2. Stop the replication job if it is still running.

   Note: This step is very important. If the replication job is still running when you try to bring up Exchange services on the secondary machine, Exchange will fail because the target files are still write-protected. The method of stopping the replication job is different depending on whether you are doing graceful migration or disaster recovery. When doing graceful migration, both primary and secondary machines are still running. You must use the Storage Replicator GUI to stop the replication job.

3. When dealing with disaster recovery, the primary (replication source) machine is not running, but the replication job on the secondary (replication target) machine may still be in a running state. In that case, you must use the Storage Replicator srTool command-line utility to cancel the replication job on the target:

   ```
   C:\program files\common files\veritas shared\vsr\srToolur.exe

   SrTool> cancel job CopyExchange –target
   ```

   Use the name of your Exchange 2000 replication job in place of CopyExchange in the last command.

Exchange 2000 — Starting the Standby Exchange Virtual Server

After the primary Exchange Server has been shut down (or stopped running in the case of disaster recovery), you may bring up Exchange services on the standby machine. On the standby machine, use Cluster Administrator to bring the Exchange cluster resource group online. This will start Exchange services, and make the IP address and network name available.

At this point, clients should be able to contact the Exchange server using the virtual server name. Since all data has been replicated from the primary to secondary server, clients should see all the same email messages, contacts information, calendar appointments and folder information they saw on the primary server.

Failback — Restoring Exchange Services on the Primary Server

In a disaster recovery scenario, there may be times when the primary Exchange server machine comes back to life. In those cases, you may want to start using the primary Exchange server again. However, while the primary server was down, there may have been substantial email activity using the standby Exchange server. The Exchange databases on the secondary server will be more up-to-date than those on the primary server. In this case, you can configure a "reverse replication" job to update the primary server's Exchange databases to match the ones on the standby server. This section outlines the steps needed to put the primary Exchange server back into production using the most current data. We will also restore the standby server to its role protecting the primary server in case of another disaster.

Creating a "Reverse Replication" Job

A "reverse replication" job is one that replicates all needed files from the standby server back to the primary server. You can create this job at any time – either as part of your original configuration tasks or when you find the need to migrate Exchange services back to the primary server. To create a "reverse replicaton" job, simply follow the steps described in the section entitled "Creating a VERITAS Storage Replicator Job to Replicate the Primary Exchange Server". The only change to make is at step 5. For a "reverse replication" job, configure the source machine name to be the name of the standby server and the target machine name to be the name of the original primary server. If you want the standby Exchange server to continue to provide service during "reverse replication", you should select the "Continue replication after synchronization" option when creating the reverse replication job. This ensures that replication captures any Exchange database I/O activity during the time replication is running. Also, when creating this job, at the "schedule" screen, select the "Clear All" button to prevent the replication job from running automatically. You will run the "reverse replication" job as a manual replication.

Note: You must not run the "forward" and "reverse" replication jobs at the same time. Also, the file name patterns you replicate should be "*.*" to replicate ALL files from the selected source directories to the target. Otherwise, the replication job may unintentionally delete files which exist in a target directory if they don't match the filename pattern.

Running the "Reverse Replication" Job

The steps to replicate data from the old standby Exchange server back to the original primary server are the same steps you would use for any graceful migration. The only change is the direction of data transfer. For failback, the replication target system is the old primary machine and the replication source is the old standby machine, which is currently the active Exchange server. As with the graceful migration scenario, the details of replication setup differ somewhat between Exchange 5.5 and Exchange 2000, due to server name restrictions.

Exchange 5.5 Failback

With an Exchange 5.5 server, you must change the computer name of the old primary server to be different than the name of the currently active Exchange server. Temporarily rename the old primary server, add it to the domain, then reboot.

After this is done, use the Storage Replicator Console GUI to start the "Reverse Replication" job to replicate the most up-to-date versions of the Exchange databases back to the original primary system. The Exchange Server can be active on the standby system while replication proceeds.

You should use the Storage Replicator Console to monitor the progress of the reverse replication job. Using the "Monitor" display, select the reverse replication job from the list of all jobs and double-click on it. This will display a dynamic window showing the progress of the replication job. When it finishes the "synchronization" phase and enters the "dynamic" phase of replication, you may shut down Exchange on the currently active server, then stop the replication job.

Finally, rename the standby server to its original name, rename the recovered primary server to its original name, then reboot both systems and start Exchange services on the original server.

Exchange 2000 Failback

With Exchange 2000 installed on both servers in a "single-node cluster" MSCS environment, the failback procedure is somewhat simpler. On the original primary server, make sure the IP address cluster resource is "offline" so Exchange services are not running. Also make sure the "physical disk" resources for that cluster resource group are online so they can receive the replication updates. On the currently active Exchange server, the entire Exchange cluster resource group should be online.

- Start the "reverse replication" job, and monitor its progress using the Storage Replicator Console. This will replicate the updated Exchange database and log files back to the original server.
  - When the replication job enters the "dynamic" replication state, use the MSCS Cluster Administrator on the active Exchange server to take the Exchange virtual server IP address offline. This will cause Exchange services to shut down on the active server.
  - Stop the replication job.
  - Finally, bring the entire Exchange cluster resource group online on the original server.

Appendix A

Graceful Migration Example

Because of the way Exchange transaction logging works, and the requirements for keeping log files and databases together, setting up VERITAS Storage Replicator with Exchange is complicated and requires a clear understanding of the methodology used for Graceful Migration to accomplish the task. Consider the following scenario:

Alice is an IT administrator for a large financial institution. Alice has been assigned to upgrade the Microsoft Exchange Server to a larger machine as part of her job. Alice wants to upgrade the Exchange server machine with as little interruption as possible to those connected to the Exchange server. Alice decides to use VERITAS Storage Replicator to perform a Graceful Migration of the Exchange server. This will be a planned service outage. Alice agrees that it's acceptable to have a small amount of downtime while the services are migrated from one server to another (known as Graceful Migration).

Events that trigger the need for Graceful Migration

During a typical business day, Alice discovers that the shared disk that is being used in the cluster is beginning to have some problems (reported by Scan Disk). Alice decides that she needs to migrate the Exchange Database now, to a different server, before it's too late. Alice wants to do this with as little interruption as possible to those connected to the Exchange server. Alice decides to use VERITAS Storage Replicator to replicate Exchange and do a Graceful Migration of the Exchange server.

Configuring Basic Services

The first step in getting VERITAS Storage Replicator working with Exchange is to make the underlying Windows NT and VERITAS Storage Replicator configuration to work.

1. Install and configure Windows NT/Windows 2000 on the Source and Target machines. If the Target machine is not a primary or backup domain controller, ensure that it will have network access to a primary domain controller or a backup domain controller at all times. It is not necessary to have identical machines; however, the disk configuration must be similar enough so that the location where the transaction logs and database files are stored on the Source server has a corresponding location on the Target server. For example, if you have a 36 GB information store on a RAID array with drive letter W, the Target server must have at least 36 GB of storage, in some form, available on the same drive letter.
2. Install VERITAS Storage Replicator on the Source and Target machines, configuring it according to the product documentation.
3. As a test of the preceding steps, replicate a small number of files from the Source to the Target. Manually verify that all data are properly replicated. It is important to verify that VERITAS Storage Replicator is operating correctly before proceeding to ensure that the Storage Replicator system and all communications between the Source and Target servers are working properly.

Installing and Configuring Exchange 5.5

Once you have verified that VERITAS Storage Replicator is working properly, install Exchange 5.5 on the Source and Target machines. Note that you should not allow Storage Replicator to mirror the Exchange installation, since this will result in incorrect information being present in the Exchange directory. If you are installing Storage Replicator to safeguard data on an existing Exchange server (and not to run Exchange on the Target server), skip step 4 below.

1. Install Exchange Server 5.5, plus the appropriate service pack, on the Source machine.
2. Run the Exchange performance optimizer (perfwiz.exe) on the Source machine. If the optimizer suggests changing the location of the transaction logs or database files, allow it to do so. At the summary screen of the optimizer, record the full path to the directory database and private and public information stores.

3. If you are installing Exchange into an existing site, verify that intra-site communications are working properly. If you are installing Exchange into a new site or organization, verify that messages flow as expected.
4. Install Exchange on the Target server, joining the same site and organization as the Source server.
5. For each connector on the Source server, create a corresponding connector on the Target server, using a higher cost for those connectors that support cost values. For example, if you have an Internet Mail Service (IMS) connector defined on the Source, create an identically configured connector on the Target. This allows messages to flow even when the Source server is offline.

Configuring the VERITAS Storage Replicator Job

The next step is to configure Storage Replicator to replicate the necessary databases:

- priv.edb
- dir.edb
- pub.edb (optional)

You may not need to replicate pub.edb, since it contains public folder information that may be present in replicas on other servers. The other two database files, however, are essential. We will replicate these files from their Source location to a different location on the Target. Note that we will move them to their proper location on the Target server at a later time. To configure the Storage Replicator job:

1. Open Storage Replicator on the Source server by clicking Start, Programs, VERITAS Storage Replicator, Console.
2. On the Configure page, click New Job. The New Job Wizard launches.
3. At the Job Type screen, select Standard (one to one) and click Next.
4. At the Job Name screen, enter a job name (e.g., ExchangeMigration) and a job description if desired and click Next.
5. At the Replication Options screen, leave the top three options checked and click Next.
6. At the Replication Pairs screen:
   a. Click Add Pair.
   b. At the Add a Replication Pair dialog box, at the Source Server field, click Select. At the Replication Servers dialog box, select the Source server and click OK.
   c. Repeat this action for the Target Server field, and click OK at the Add a Replication Pair dialog box.
   d. Click Next.
7. At the Replication Rules screen, leave Source Tree selected in the View As list. Expand <Source server>, and navigate to the Exchange folder (Exchsrvr).
   a. Select the Exchsrvr folder and click Add Rule.
   b. At the Rule screen, in the Inclusions and Exclusions section, click Add.
   c. At the Inclusion/Exclusion dialog box, select Inclusion (default), and, in the Filter field, type priv.edb. This will cause Storage Replicator to replicate only this file. Be sure that Apply to Subdirectories is checked. Click OK.

      Note: If you want to replicate pub.edb as well as priv.edb in the Filter field, type *.edb. This will cause Storage Replicator to replicate both database files, as well as dir.edb (see step e.).
   d. In the Target Path section, verify that the default Target path is large enough for the priv.edb file and click OK.
   e. If you are replicating both pub.edb and priv.edb, skip to Step 8 (the next few steps would be redundant). If you are only replicating priv.edb, click Add Rule again.

f. At the Rule screen, in the Inclusions and Exclusions section, click Add.

g. At the Inclusion/Exclusion dialog box, select Inclusion (default), and, in the Filter field, type dir.edb. This will cause Storage Replicator to replicate only this file. Again, be sure that Apply to Subdirectories is checked. Click OK.

h. In the Target Path section, verify that the default Target path is large enough for the dir.edb file, and click OK.

i. Click Next.

8. At the Replication Schedule screen, leave the default schedule and click Finish.
9. At the Configure page, notice your new job in the list. We will run the job later.

Graceful Migrations

For a Graceful Migration, the Source server may not contain public folders. If it contains connectors, those connectors will have to be re-created because they cannot be migrated. Note that it is not necessary to rename the Target server so that it has the same name as the Source server; the steps below ensure that clients can still connect to the original server and that their mailbox profiles will automatically be updated to point to the correct location for their mailbox. Note that this process minimizes, but does not eliminate, downtime; for that reason, it is probably best used for circumstances where you can schedule the migration during a period of low user activity.

Steps to Take Before the Migration

1. If the Source server was the first server installed in the site, follow the steps in Microsoft Knowledge Base article Q152959 to rehome that server's site-specific functions to another server in the site (the Target server, or any other in the same site).
2. If there are any public folders replicas on the Source server, remove them. If there are any public folders whose only replica is on the Source server, create replicas on another server (but not the Target server) and allow time for replication to occur.
3. Create a distribution list that contains all the users on the Source server. (In general, this type of DL is useful for routine maintenance, so you might consider creating one DL for each server in your organization.)
4. Use Exchange Administrator in raw mode to locate the distinguished name (DN) of the System Attendant mailbox and the mailbox you normally use for sending administrative messages to users. See item c below for the distinguished name format.

   a. Run Exchange Administrator in raw mode by using the admin.exe /r command. Note that it is extremely dangerous to make changes in raw mode, so you should be very careful.

   b. Locate the object whose DN you're looking for. For the system attendant, look in the container named for your server. (It's in the Servers object underneath the Configuration subnode of the Site object.)

   c. Use the File | Raw Properties command to see the object's properties. Locate the Obj-Dist-Name attribute. For the system attendant, the name will usually be of the form /o=organization/ou=siteName/cn=Configuration/cn=Servers/cn=sourceServerName/cn=Microsoft System Attendant.

Steps to Take at Migration

If the Source server was the first server installed in the site, follow these steps

1. On the Source server, restrict logons to the information store so only the system attendant mailbox and your administrative mailbox can log on. This allows you to keep users from logging on to their mailboxes while you're migrating. To do this:

   a. Open the Registry Editor (use regedt32 by clicking Start, Run, and typing regedt32).

   b. Open the key HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\MSExchangeIS\ParametersSystem.

c. Add a new REG_MULTI_SZ value called "Logon Only As." On separate lines, enter the DN of the system attendant mailbox and your administrative mailbox:

i. Click Edit, Add Value.

ii. In the Add Value dialog box, enter Logon Only As in the Value Name field, and select REG_MULTI_SZ from the Data Type list.

iii. In the Multi-String Editor dialog box, type the distinguished name of the system attendant mailbox, hit Enter, then type the distinguished name of your administrator mailbox.

iv. Click OK.

d. Stop and restart the information store service. This has the desirable side effect of committing any pending transactions in the log files.

2. Stop the IS service on the Source and Target servers. Verify that the Source server shut down cleanly by checking the event log and using the eseutil /mh command against priv.edb. If there are no errors in the event log, and if eseutil reports that the database is consistent, it is safe to proceed. If there are errors, consult your Microsoft Exchange documentation.
3. You must now run the Storage Replicator Job. At the Configure page of the VERITAS Storage Replicator Console, rightclick the job and click Start Now.
4. When replication is complete, use the eseutil /mh command to verify that the Target database is consistent.
5. Move all data from the mdbdata files on all local drives of both servers to a safe location outside the exchsrvr folder tree.
6. Restart the IS service on both servers. This creates empty public and mailbox stores.
7. Using the administrative mailbox you chose before beginning the migration, log on to the Source server. Create a message to the DL you created earlier; the message should say something like this:

   We are moving mailboxes from sourceServer to targetServer for scheduled maintenance. Your mailbox items are safe, but are unavailable right now. Please log off so that we can complete the move procedure. Your data will be available again after time. Any messages that you send during this session will not be delivered, and you cannot receive mail right now. Personal appointments or other items that you create in this session will be lost after the move. You may direct any questions to administratorName.
8. On the Source server, remove the Logon Only As Registry key you created in step 1; stop and restart the IS. This allows users to log on, and they'll see the message you created in step 7.
9. On the Source server, use the Exchange Administrator application's Tools | Move Mailbox command to move all mailboxes from the Source server to the Target server. This step is necessary to update the directory's knowledge of where mailboxes live. Because the mailboxes contain only the message you sent in step 7, the move will be much faster than normal.
10. On the Target server, restrict logons, using the procedure from step 1. Note that your mailbox will have the same DN on the Source and Target servers, but that the system attendant's DN will be different.
11. Stop the IS service on the Target server. Move the replicated copy of priv.edb (and pub.edb if applicable) to the proper location (exchsrvr\mdbdata on whichever physical disk you've configured to hold the database). Move the replicated copy of dir.edb to the proper location (exchsrvr\dsadata).
12. On the Target server, run the isinteg – patch command to restore GUIDs to the private store. When the patch process completes, start the IS service.
13. Verify that the Target server's IS started normally and that all data are present:

    a. Check the event log for error or warning messages.

    b. Use the Mailbox Resources view in Exchange Administrator to estimate whether the mailbox sizes appear to be correct.

    c. Log on to the administrative mailbox on the Target server; verify that you can access mail in that mailbox.

14. On the Target server, remove the Logon Only As key to re-enable client logons. Stop and restart the IS.
15. If you want to use connectors homed on the Target server, recalculate routing on the Target server to force the use of the new connectors.

Steps to Take After the Migration

Once the migration is complete, it is important to keep the Source server running until each client, whose mailbox was on the home server, has logged in at least once. If this is prohibitive, clients can reconfigure their profiles to point to the new server.

Appendix B

Disaster Recovery Example

When we consider how to provide Disaster Recovery for Microsoft Exchange Servers, we understand that what is needed is a way to provide for fast recovery of an Exchange Server that is not able to perform as intended. Sometimes the Exchange Server becomes corrupt and unusable, other times the operating system crashed or the network connection is down. Whatever the case, there should be a way to recover from a downed/unavailable Exchange Server. Using VERITAS Storage Replicator, the Exchange database can be replicated to a different server at any location where it can be made available at the needed time to take over for the faulty primary Exchange server. The following example shows how to replicate the Exchange database and required files to provide relief in the case of a disaster.

VERITAS Storage Replicator Exchange Replication for Disaster Recovery

Tom is an IT administrator for a midsize law firm, who is assigned to take care of the Microsoft Exchange Server. As part of Tom's job, he is required to keep a backup copy of the Exchange Database and associated files in case of a disaster. Tom has been doing this with backup software successfully for some time. One of the things that concerns Tom is the recovery time for the Exchange Server. For that reason, the server that is hosting Exchange has been clustered for a greater level of protection.

Events Leading Up To a Disaster

During a typical business day, Tom is notified that the Exchange server is down. The hard drive that was housing the Exchange database has failed and the disk array is not accessible. Apparently the problem spans beyond a single disk failure within the array. Tom is faced with a serious problem that goes all the way to the attention of the president of the company. Tom must rebuild the Exchange database from backup tapes. Tom is faced with considerable downtime and his boss is wondering why they are not back online now. After this experience, Tom decides to use VERITAS Storage Replicator to replicate the Exchange database to another server so that it can be brought back online faster than recovery from tape. Tom uses the best-practices procedure for replicating Exchange for disaster recovery recommended by VERITAS. Now Tom feels much more confident that he can handle a disaster that affects Exchange.

Best Practices Using VERITAS Storage Replicator and Exchange 5.5 for Disaster Recovery

Exchange 5.5 Installation

Microsoft Exchange needs to be installed on both the primary server and secondary server for disaster recovery. The directory database, "dir.edb" file, is machine-specific. To get around this issue Exchange 5.5 will be installed in the following manner.

At the beginning of the installation one server is named SERVER1, and the other server is named SERVER2. For this example, SERVER1 will act as the original Exchange server.

1. Install Exchange 5.5 to an NTFS-formatted drive on SERVER1. The installation directory must be the same on both machines, as well as the account that was used to install Exchange.
2. After Exchange is installed, run Optimizer and allow the files to be moved automatically. The log files may be moved to another location for better performance. For example, the "Information Store" database file may be placed on G:\exchsrvr\mdbdata while the "Information Store" log files may be placed on J:\exchsrvr\mdbdata. Also, the "Directory Store" log files may be placed on different disks from the "Directory Store" database file.
3. Now, on SERVER1 apply the latest service pack for Exchange 5.5 (currently SP4).

4. Now, stop the Microsoft Exchange System Attendant service on SERVER1. This will also stop all other Exchange services on SERVER1 so they don't interfere with the installation of Exchange on SERVER2.

   At this point, Exchange 5.5 server has been successfully installed on SERVER1. Now we must install Exchange 5.5 server on SERVER2. To get SERVER2 to act as a disaster recovery server for SERVER1, SERVER2 must masquerade as SERVER1 during Exchange 5.5 product installation. This will ensure that various registry keys related to Exchange server operations are set up to allow SERVER2 to act as if it were SERVER1 after a disaster failover. To accomplish this, do the following steps on SERVER2:

5. Use the registry editor to change the **ComputerName** value associated with the registry key named HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\ComputerName\ActiveComputerName. Set the value of the key to SERVER1.
6. Install Exchange 5.5 on SERVER2. Make sure to use exactly the same installation directory and service account name as you did during the installation on SERVER1.
7. Run the Optimizer on SERVER2. When it is done, make sure all selected paths exactly match the ones used on SERVER1.
8. Apply the latest service pack for Exchange 5.5 on SERVER2.
9. Finally, change the **ComputerName** setting described in Step 4 back to its original value.

VERITAS Storage Replicator Installation

Install VERITAS Storage Replicator on each server that will host Exchange. This requires that one of the servers be installed with the Replication Management Service (RMS), or if there is an existing Storage Replicator installation and an RMS is already set up, then the existing "Replication Neighborhood" will need to be specified for the new Storage Replicator installation. Each new server that is installed with Storage Replicator (made available for replication) will need to specify the name of the "Replication Neighborhood" for the RMS to be able to communicate with the server and direct the replication process.

Once the installation is complete for the servers that will host Exchange, the system needs to be tested to ensure that replication between the servers is working properly. This is done by creating and running a small (number of files to be replicated) standard job. Then monitor the job using the Storage Replicator console. After the job is complete, verify that the data has been replicated correctly by reviewing the files on the Target server.

Replication Job

Once VERITAS Storage Replicator™ has been installed, we can set up replication. For disaster recovery, we set up one continuous replication job to copy all Exchange data and log files from SERVER1 to SERVER2. This ensures that all needed files are replicated, and we can quickly bring up the standby server in case of a disaster. Because the job is set up as a continuous replication, changes to Exchange data and log files on SERVER1 will be quickly propagated to the files on SERVER2. Thus, SERVER2 will have data that is almost as up-to-date as the data on SERVER1. In the event of a disaster affecting SERVER1, some of the data may not have been replicated to SERVER2. But normal Exchange server startup procedures and the use of the Exchange isinteg –patch command ensure the resulting replica is consistent.

Creating the Replication Job

1. Configure a single replication job, selecting the following options:
   - No Changes on Target
   - Exact Replica on Target
   - Continue Replicating after Synchronization

   You may also select the "Prescan" option if desired.
2. Select SERVER1 as the replication source, and SERVER2 as the replication target.
3. Now, configure the replication rules to replicate all source directories containing Exchange data files.

4. For each source directory, create a rule to select "*.*" to copy all files.
5. For each target directory, specify a "Custom Path" that exactly matches the source path name.
6. When creating a replication schedule, use the "Set All" button to allow replication to proceed at all times.
7. If there are any Exchange services running on SERVER2 as the result of installation, stop them. This includes the Exchange System Attendant, Exchange Directory Service and Exchange Information Store.
8. Start the replication job.
9. Finally, restart Exchange services on SERVER1 to put it into production as an Exchange server.

The initial synchronization may take some time as Exchange database files are copied from SERVER1 to SERVER2. After Storage Replicator copies the initial Exchange database and log files from SERVER1 to SERVER2, it will continue to replicate changes to those files, keeping an up-to-date copy of all relevant database files on SERVER2.

Validating the Solution

To test the system, a simulation of activity is run on the Exchange server. The Exchange server will be hit with Mailstorm at the rate of one e-mail per second (1 MB in size) to various e-mail boxes. Some e-mails will have attachments. Public folders and posts in those folders will also be created.

Bringing Exchange up on the Target server

Next, we need to bring down the primary server and remove it from the Domain. Then rename the Target server to SERVER1 and add it to the domain, then reboot. The Directory Service, MTA and System Attendant should come up without errors in most cases. If they do have errors, you may want to run eseutil /p on the dir.edb (which takes only a few seconds to a couple of minutes depending on the size of the database). One of the ways you might get errors is if you had a power failure and lost connectivity with the Target before the system could safely shut down. The Information Store will most likely generate an 1011 Error when trying to start the service on the Target server. This requires running `isinteg –patch`, which should taken less than 10 seconds on databases that ranged from about 8.5 to 11.5 GB. Once this portion is complete, the IS should come up without error. All public folders and posts should be seen as well as messages. Any changes (limits, etc.) to folders and mailboxes should also be seen on the Target server.

Automating Changes Required on the Target Server

Below is a batch file used to rename Target computer and add to domain. This will rename the Target computer to the Source computer's name. Then it will delete the Source from the domain and add the Target, which now has the Source name, to the domain. Requires the NT Server Resource kit for netdom.exe and shutgui.exe. All values within the asterisks, "**", will need to be changed for each environment.

`c:\winnt\regedit.exe -s D:\exchsrvr\rename.reg` (contents below) — to rename the Target computer.

`c:\netdom.exe /domain:**domain** /user:**domain**\**domainadmin** /password:**password** member **computername** /delete` — deletes Source.

`c:\netdom.exe /domain:**domain** /user:**domain**\**domainadmin** /password:**password** member **computername** /joindomain` — adds Target.

`c:\shutgui.exe /l /r /t:00 /c` — reboots Target.

The rename.reg file that is referenced above contains the following:

```
REGEDIT4

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\ComputerName\ComputerName]

"ComputerName"="SERVER1" — name should be the Source computer's name.

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\ComputerName\ ActiveComputerName]

"ComputerName"="SERVER1" — name should be the Source computer's name.
```

The above batch file should be run immediately on the Target server when the Source goes down. The process of troubleshooting Exchange on the Target should begin after the reboot. As stated above the Target machine was consistently getting an 1011 Error when trying to start the Information Store. This required running isinteg – patch and took less than 10 seconds each time.

Best Practices Using VERITAS Storage Replicator and Exchange 2000 for Disaster Recovery

Exchange 2000 uses the directory information maintained on a Windows 2000 server running the Active Directory service. To prepare for disaster scenarios, the computer providing Active Directory services should be different from the primary and secondary Exchange servers. Additionally, you should use Active Directory replication tools to ensure there is a copy of the directory information in case the primary Active Directory system fails.

Exchange 2000 Installation

As in the Exchange 5.5 example, the primary server is named SERVER1 and the secondary is named SERVER2. But for Exchange 2000, we use an Exchange virtual server name of EMAILSRVR. The steps to set up the disaster recovery configuration are as follows:

1. Install Microsoft Cluster Server software on both SERVER1 and SERVER2. These machines are not part of the same cluster. Install them as separate clusters.
2. Configure an Exchange cluster failover group with an IP address resource an a network name resource with a name of EMAILSRVR.
3. Install "cluster aware" Exchange 2000 server on SERVER1.
4. Create an Exchange System Attendant cluster resource and make it a member of the Exchange cluster failover group.
5. Use Cluster Administrator on SERVER1 to put the Exchange cluster resource group IP address offline. This will also stop Exchange services on SERVER1.
6. Repeat steps 2 through 5 on SERVER2, configuring the Exchange cluster resource group to have the identical physical disks, IP address, and network name as SERVER1.
7. Put the Exchange cluster resource group back online on SERVER1. Clients can now use the virtual server named EMAILSRVR for Exchange services.

VERITAS Storage Replicator Installation

The installation steps for VERITAS Storage Replicator™ are the same for Exchange 2000 support as they are for Exchange 5.5, described in the previous section.

Replication Job

Use the Storage Replicator Console to create a single continuous replication job to replicate all Exchange 2000 data and log files from SERVER1 to SERVER2. When this job is running it will propagate all data and log file changes from SERVER1 to SERVER2.

Configuring the Replication Job

1. Configure a single replication job, selecting the following options:
   - No Changes on Target
   - Exact Replica on Target
   - Continue Replicating after Synchronization

   You may also select the "Prescan" option if desired.
2. Select SERVER1 as the replication source, and SERVER2 as the replication target.
3. Now, configure the replication rules to replicate all source directories containing Exchange data files. For example, if Exchange data files are in the G:\EXCHSRVR path and log files are in the J:\EXCHSRVR path, set up two rules – one to replicate each directory tree.
4. For each source directory, create a rule to select "*.*" to copy all files.
5. For each target directory, specify a "Custom Path" that exactly matches the source path name.
6. When creating a replication schedule, use the "Set All" button to allow replication to proceed at all times.
7. If there are any Exchange services running on SERVER2 as the result of installation, stop them by using Cluster Administrator to take the Exchange cluster resource group's IP address offline. This will stop all Exchange services for the virtual server named EMAILSRVR.
8. Start the replication job.
9. Finally, on SERVER1, use the Cluster Administrator to bring the Exchange cluster resource group online. This will restart Exchange services on SERVER1 to put it into production as an Exchange server.

As with the Exchange 5.5 scenario, the initial file synchronization may take some time as Exchange database and files are copied from SERVER1 to SERVER2. After Storage Replicator copies the initial Exchange database and log files from SERVER1 to SERVER2, it will continue to replicate changes to those files, keeping an up-to-date copy of all relevant database files on SERVER2.

Bringing Exchange 2000 up on the Target server

If SERVER1, the primary Exchange 2000 server, dies, you can bring up Exchange 2000 services on SERVER2, the secondary server, providing fast recovery of Exchange services to clients. Since the replication job was doing continuous replication, the data on SERVER2 will be almost up-to-date with that which was present on SERVER1. Of course, a few messages may have been lost in transit when SERVER1 died. However, Exchange 2000's database technology and use of transaction log files ensures the resulting replica database can be put into a consistent state. To bring up Exchange 2000 services on SERVER2, use the following steps:

1. Use the Storage Replicator **srTool** utility to stop the replication job on SERVER2, the replication target machine:

   ```
   C:\program files\common files\veritas shared\vsr\srToolur.exe

   SrTool> cancel job CopyExchange –target
   ```

   Use the name of your Exchange 2000 replication job in place of *CopyExchange* in the last command.

Note: This step is extremely important. If the replication job is not stopped in exactly this way, the target files will continue to be marked read-only, and Exchange 2000 services startup will fail.

2. Use the Cluster Administrator to bring the Exchange cluster failover group online on SERVER2. This will make EMAILSRVR available as an Exchange 2000 virtual server, and clients may reconnect to it.

   With Exchange 2000, it is not necessary to run the isinteg –patch command, since Exchange 2000 server performs the same work itself internally. There is no reboot involved when using this method of providing a virtual server name.

Appendix C

Accounting for Data Corruption

Because of the way Exchange transaction logging works, and the requirements for keeping log files and databases together, the process of setting up VERITAS Storage Replicator with Exchange is complicated and requires a clear understanding of the methodology used to replicate Exchange. In the event that the database becomes corrupt, additional steps should be taken to ensure that recovery can be made from a previous copy of the database. This can be done by using a script file to create a copy of the Exchange database periodically, or by manually creating a copy of the database and running eseutil /p or isintegp against the database to validate it for consistency.

- The /r (recovery) switch does a nondestructive recovery. It replays any outstanding log files and tries to fix missing page links, etc., as best as it can without truncating any pages. It's always safe to run this, but it is not guaranteed to make the database consistent — it's a best-effort kind of fix. MS describes this as "performing recovery, bringing all databases to a consistent state."
- The /p (repair) switch is used to "fix a corrupted or damaged database." It's like a tree surgeon: If anything is corrupted, it gets cut off. That means that if eseutil finds orphaned pages, missing links, bad crosslinks, etc., it will happily truncate pages. Depending on which pages get truncated, this may have minimal (Joe User loses one message) or major (everyone loses all attachments) effect. You only run this as a very, very last resort after exhausting all other avenues.

Appendix D

References

- Microsoft Knowledge Base article Q237767, "XADM: Understanding Offline and Snapshot Backups" http://support.microsoft.com/support/kb/articles/Q237/7/67.asp
- Microsoft Knowledge Base article Q155216, "XADM: How to Move Exchange Server to a New Computer with the Same Name" http://support.microsoft.com/support/kb/articles/Q155/2/16.asp
- Microsoft Knowledge Base article Q199954, "XADM: Forklifting All Users to a New Server" http://support.microsoft.com/support/kb/articles/Q199/9/54.asp
- Microsoft Knowledge Base article Q152959, "XADM: How to Remove the First Exchange Server in a Site" http://support.microsoft.com/support/kb/articles/Q152/9/59.asp
- Microsoft Knowledge Base article Q146764, "XADM: Limiting Logons to the Information Store" http://support.microsoft.com/support/kb/articles/Q146/7/64.asp

VERITAS®

WHITE PAPER

VERITAS Software Corporation
Corporate Headquarters
350 Ellis Street
Mountain View, CA 94043
650-527-8000 or 866-837-4827

For additional information about VERITAS Software, its products, or the location of an office near you, please call our corporate headquarters or visit our Web site at www.veritas.com.

sales@veritas.com

02-20576