

Backup and Restore of **Microsoft Exchange 2000**

VERITAS Windows Platform Team

Understanding backup and restore processes allows administrators to implement effective contingency plans for outages or disasters. This article describes the interaction among elements in a Microsoft® Exchange® system and highlights the use of VERITAS NetBackup™ for Exchange to simplify regular backups, restore important data, and minimize disruption.

Any computing environment needs the ability to recover data or an entire system following outages or disasters. Regular backups significantly contribute to a successful recovery. The Microsoft® Exchange® 2000 backup and restore processes provide mechanisms for maintaining system continuance and minimizing disruption to users in the Exchange environment. VERITAS NetBackup™ for Exchange simplifies backup and restore of Exchange 2000 databases and mailboxes.

Understanding the Microsoft Exchange 2000 environment

Understanding the interaction between Microsoft Exchange 2000 and the Microsoft Windows® 2000 Active Directory® eases maintenance and recovery of Exchange 2000 environments. Likewise, understanding the function of database files, transaction logs, and patch and checkpoint files helps administrators to restore data from failure or from a particular point in time.

Active Directory and Exchange 2000 Server

In Active Directory, the domain or the forest as a whole owns all objects. If a single domain controller is removed, no objects from the directory are lost (unless the server is the

last domain controller that exists for a domain). The Active Directory objects exist as copies on the remaining domain controllers; each domain controller in the domain is a complete backup of all the others.

Exchange 2000 requires access to the objects stored in the Windows 2000 Active Directory. Therefore, regular backups of the Active Directory and a redundant domain controller are critical to the survival of an Exchange 2000 environment. Sometimes administrators may need to access out-of-date information, such as when they inadvertently delete an important container or install a rogue application. In such cases, earlier copies of Active Directory can

help restore the information. Active Directory provides authoritative restoration capabilities.

Even the smallest Windows 2000 environments should have at least two domain controllers to provide two replicas of the Active Directory database. Larger organizations should have no fewer than three domain controllers per domain for redundancy.

Good practice dictates that backup copies of the Active Directory for each domain be stored in a safe off-site location. At minimum, perform at least one backup for each domain before and after making significant changes to Active Directory, such as when installing Exchange 2000.

Regular backups of the Active Directory and a redundant domain controller are critical to the survival of an Exchange 2000 environment.

More than one replica of Active Directory benefits disaster recovery efforts. Destruction of one of the directories does not interrupt service to clients; therefore, the restoration process is not an emergency. Administrators also gain several recovery options. They can restore from a backup, rebuild the server and join the server to the domain again as a domain controller, or bring up a third server as a replacement domain controller.

Exchange 2000 database files

Microsoft Exchange 2000 accommodates up to 20 database stores, each consisting of two database files identified by an edb extension.

During normal operation, the database file itself is never up-to-date. The store service manages a large in-memory buffer of store data and periodically flushes modified pages to disk. However, this approach creates a lag between normal activity in a database file and updating the database file on disk. This lag endangers database integrity if a sudden system failure occurs.

Dumping the data into transaction log files secures the modified pages to disk. This technique is much faster than updating the database (which incurs the overhead of updating multiple indexes, random disk access, and so forth) and allows Exchange to deliver consistently high performance—even under heavy loads.

When the store service terminates gracefully, it writes all modified pages in the database buffer to the database file, bringing the file to a consistent state before the service shuts down. If the store service terminates ungracefully (crashes), the database remains in an inconsistent or unknown state, but the transaction logs contain all the information required to restore the database. Replaying these logs against the database returns it to a consistent state—as it would achieve after a clean shutdown.

Exchange 2000 streaming database

The streaming database is a new feature in Exchange 2000. In Exchange 5.5, every message that arrived from the Internet was converted to a Messaging Application Programming Interface (MAPI) format, which required converting Multipurpose Internet Mail Extensions (MIME) content into a format that the Exchange database could index, manage, and recognize. Exchange 2000 stores incoming Internet messages in the streaming database. This file has an stm extension and is a companion to the edb database file.

The stm file contains raw content; it does not include indexes or properties. When a message arrives, Exchange 2000 promotes only the indexing and the management information into the edb file. Exchange 2000 then references locations in the stm file where users can go to read that message. This approach allows faster transmission of Internet messages and reduces conversion of message formats. The edb and stm files are a matched set and

should be treated as a single file. If an administrator loses the stm file while backing up the edb file, the edb file is useless.

Exchange 2000 transaction logs

The Microsoft Exchange database uses transaction logs to accept, track, and maintain data. All transactions are first written to transaction logs and memory, and finally to the database. Transaction logs can help to recover Information Store databases after database failure or corruption.

BEST PRACTICES FOR BACKING UP MICROSOFT EXCHANGE 2000 ENVIRONMENTS

Administrators must determine where and how to store Exchange data to ensure the best possible performance and eliminate single points of failure. Key considerations include the following:

- ▶▶ Mirror the system drives.
- ▶▶ Use a RAID-5 (striped set with parity) for the Information Store; this approach will improve performance and add an extra layer of protection against hardware failure.
- ▶▶ When allocating space for the Information Store, consider using twice the amount of the expected Store size. This space will be used when running database maintenance (ESEUTIL), which creates a temporary database of the same size. Redirection of the temporary database to another partition (if sufficient space is not available) can degrade the performance of ESEUTIL.
- ▶▶ Place the individual stores (dir.edb, priv.edb, and pub.edb) and their respective transaction logs on different disks to eliminate a single point of failure.
- ▶▶ Disable circular logging; this action will allow the roll back of transaction logs from any point in time and enable incremental and differential backup modes.
- ▶▶ Performing scheduled backups is the only real way to protect data. The length of the backup window will determine the level of backup performed. A schedule based on the time restrictions should be devised to allow for daily backups. Microsoft recommends daily, full online backups of the Exchange Server. If time is a constraint, use a combination of full and incremental or differential backups. For example, the weekend might provide a larger backup window and a good opportunity to do full backups. During the week, an incremental or differential backup can be performed during more restrictive windows.

The Information Store has two separate databases, but transaction logs are kept in a single set. Because transactions are first written to the edb.log file and then later written to the database, the actual database can be a combination of the uncommitted transactions in the transaction log file and the actual edb database file. When the edb.log file is filled with data (approximately 5 MB), it is renamed and a new edb.log file is created. When the edb.log file is renamed, the renamed log files are stored in the same subdirectory. The renamed log files are named in a sequential numbering order (for example: edb00014.log, edb00015.log, and so forth using hexadecimal).

Transactions contained in the log files are committed to the respective edb file when the service shuts down normally. Log files should not be manually purged; it is best to purge logs through the backup process.

Together with the appropriate Exchange online backup tapes, the transaction logs enable administrators to roll forward the day's transactions and restore a database without losing messages.

Transaction logs have no long-term value; they are useful only if they were created since the most recent online backup. There is no reason to back up or restore the transaction log files independently of the databases to which they relate.

Transaction logs during a backup. Committed transaction logs are truncated (deleted) by Microsoft Exchange during backup. These logs are no longer required since they have been committed to the database file and they have been written to the backup media. If the transaction-log volume becomes corrupt, administrators will lose the ability to roll forward after a database restore.

Transaction logs during a restore. Administrators have two choices during the recovery of the Exchange 2000 database.

First, they could keep the existing transaction logs and overwrite any transaction logs that exist. After file restoration and service startup, the database will commit the transactions in the restored logs. If contiguous logs exist on the server beyond the log with the highest number restored, those transactions will also be committed. If any gap occurs in the numeric sequence of log names, no further transactions will be committed beyond the gap.

This scenario is useful when the transaction logs are intact but the database requires restoration. By keeping existing transaction logs, Microsoft Exchange Server can recover to the point of the failure instead of the time of the last full or incremental backup (differential incremental backup or cumulative incremental backup).

When the store service terminates gracefully, it writes all modified pages in the database buffer to the database file, bringing the file to a consistent state before the service shuts down.

The second option is to delete the existing transaction logs. Several situations, such as restoring the Information Store to an alternate server, restoring to a previous date without recommitting all the logs that are still on the disk, or performing a full restore, require the deletion of existing transaction logs.

Exchange 2000 database patch files

Database patch files handle transactions written to the database during a backup. If a transaction causes an update to a part of the edb file that has already been backed up, then it is written to the patch file for that database. Patch files exist only during the backup process. During the Microsoft Exchange Server recovery process, patch files update the restored database file with the transactions that were in progress during the backup.

Checkpoint files

Checkpoint files recover, or play, data from transaction logs into edb files. The checkpoint is the place marker in the edb.chk file that indicates which transactions have been committed. Whenever data is written to an edb file from the transaction log, the edb.chk file is updated with information specifying that the transaction was successfully committed to the respective edb file. During recovery, Exchange determines which transactions have not yet been committed by reading the edb.chk file or by reading the transaction log files directly (in which case edb.chk is not required).

The Information Store and Directory Service maintain separate edb.chk files, which they read during startup. They then use the transaction logs to play any uncommitted transactions into the edb files. For example, if an Exchange Server experiences an outage and transactions have been recorded into the transaction log but not to the database file, Exchange attempts recovery on startup by automatically recording transactions from the logs to the database files.

Backing up Exchange 2000

Administrators can back up all Exchange databases on an Exchange server or a specific set of databases or storage groups. Administrators also can choose to restore all the messaging information on an Exchange server or to restore an individual database or storage group.

Although administrators can back up all databases individually, backing up an entire storage group at one time is recommended. Individual database backup will require multiple backups of the log files, and when backing up and restoring Exchange 2000 databases,

administrators cannot run multiple backup or restore processes in a single storage group.

The following steps occur during a full backup:

- ▶▶ Write database files to the backup media.
- ▶▶ Create patch files to accommodate updates to the database during the backup.
- ▶▶ Write transaction logs to the backup media.
- ▶▶ Write patch files to the backup media.
- ▶▶ Truncate (delete) committed transaction logs.

Figure 1 illustrates the process of backing up an Exchange 2000 database.

Backup types

VERITAS NetBackup can perform full, copy, incremental, and differential backups. The data's importance determines the type of backup. Each backup type offers advantages and disadvantages in terms of data storage, performance, and time requirements. The two general backup methods are online and offline.

Online backups. An online backup allows the database to continue running during data backup. An online backup does not affect users or interrupt processing. An online backup can be either partial or full. Full backups copy everything in the database, and partial backups copy only the log files.

An online, full backup is the preferred backup type. A full backup copies Exchange database files and Exchange log files. It

deletes transaction log files that contain transactions committed to the server database. Restoration from a full backup usually involves only one backup tape.

Offline backups. An offline backup lets administrators save a copy of the database without copying the log files. An offline backup is always the second choice, however. Administrators must dismount the database before performing the backup, which prevents users from sending or receiving mail during the process.

Exchange 2000 configuration data backup

In addition to backing up the Exchange databases, administrators should back up all mission-critical data such as the contents of users' mailboxes and the configuration data necessary to run the Exchange servers. Administrators also should routinely back up Exchange 2000 configuration data, including the following:

- ▶▶ Web storage system databases and supporting files
- ▶▶ Active Directory
- ▶▶ System state, including the Microsoft Internet Information Services (IIS) Metabase, which contains the protocol information Exchange 2000 uses

Backup verification and validation

The ability to restore data and servers depends on the quality of the backups; therefore, it is important to verify the success of a backup procedure. For complete fault tolerance, verify a backup procedure at the event and data levels.

Restoring Exchange 2000

Different disasters can afflict an Exchange environment. Administrators might need to restore a deleted or damaged mailbox, for example, or they might need to restore one or more databases or storage groups. If a major disaster occurs, administrators might need to restore an entire Exchange server using the /disasterrecovery switch. This procedure includes ensuring that Active Directory is still intact, restoring the Exchange 2000 system state and restoring all the Exchange databases.

Administrators can use VERITAS NetBackup to restore damaged databases, storage groups, or the entire Exchange server from backup.

Exchange 2000 database recovery

A restore plus transaction roll-forward is the easiest and sometimes only way to recover the Exchange 2000 databases after a corruption. Figure 2 illustrates the process of restoring an Exchange 2000 database.

Administrators should not restore both Microsoft Exchange Mailbox and Microsoft Exchange Server objects at the same time. If an administrator shuts down Exchange services to restore the Exchange Server databases, the restore of the Mailbox objects will

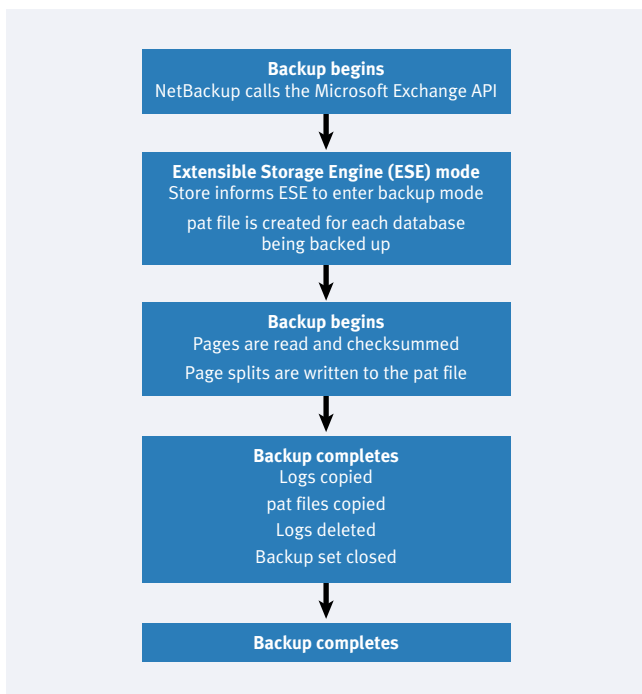


Figure 1. Exchange 2000 database backup flow

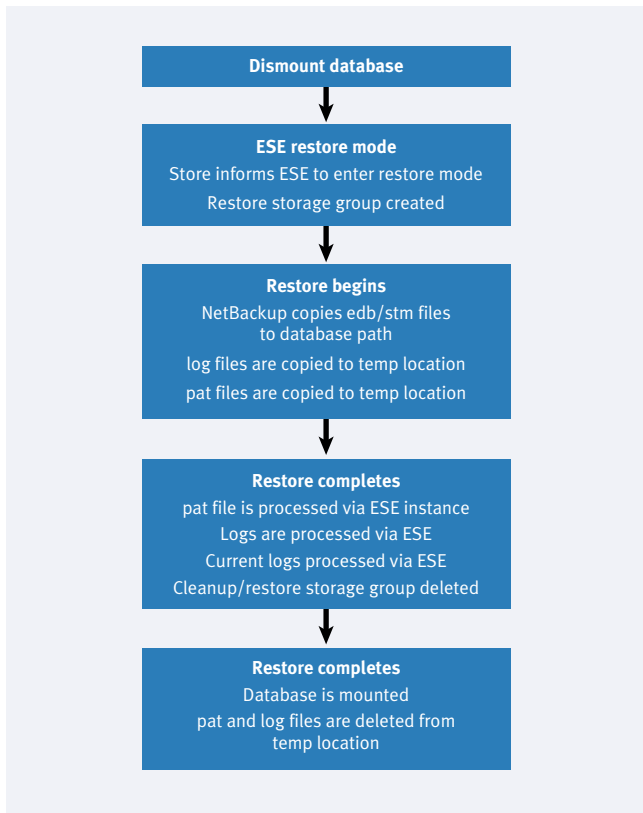


Figure 2. Exchange 2000 database restore flow

fail. Or, if the restore of the Exchange Mailbox items finishes before the restore of the Exchange databases starts, restored databases will wipe out the restored Mailbox objects.

Multiple database recovery in a single storage group

Administrators can run only one backup or restore process on a particular storage group at a time. Multiple simultaneous backup or restore processes must be run on multiple storage groups. Therefore:

- ▶ Administrators can restore only one backed-up database at a time in a given storage group.
- ▶ Administrators can restore backups of multiple storage groups simultaneously.

When restoring multiple databases in the same storage group, first mount the just-restored databases before proceeding with the next database restoration. For example, databases A, B, and C are in the same storage group and were backed up together. To restore database A, set only database A for restoration. If an administrator then

decides to restore database B, database A must finish restoring first and then be mounted. Once database A has been mounted, an administrator can start restoring database B. To restore both databases simultaneously, an administrator must originally mark both database A and database B for restoration.

The restore process creates a Restore.env file in a temporary directory and removes the file when the database has been successfully mounted. If an administrator chooses to restore databases A and B together, the Restore.env file will contain information for both. However, when an administrator restores databases individually, the Restore.env file for database A must finish processing before the restore process for database B can begin.

Full server recovery

A full server recovery involves restoration of Exchange 2000 Server or Windows 2000 Server or both. Multiple storage groups and databases in Exchange 2000 complicate the restoration process.

When performing a full server recovery, administrators must first reinstall Microsoft Exchange 2000 Server in Disaster Recovery mode before recovering Exchange data. Disaster Recovery mode retrieves information from Active Directory to correctly reinstall Exchange.

Administrators should know the procedures required for recovering different types of servers. Exchange 2000 servers can be dedicated to specific roles, such as running Key Management Server (KMS) or Site Replication Service (SRS). These can be rebuilt after the Exchange server is up and running.

Be aware that the mode in which an Exchange server is running might require additional steps for recovery. For example, recovering an Exchange 2000 cluster server requires more steps than recovering a single Exchange 2000 member server.

Using NetBackup for Exchange

VERITAS NetBackup utilizes the Microsoft Exchange APIs to perform online backups of the Microsoft Exchange Information Store and Directory along with all associated transaction log files. NetBackup supports advanced Exchange roll-forward

and roll-back operations so administrators can restore Exchange databases to any point in time, a key requirement of many large-scale Exchange environments.

To use NetBackup for Microsoft Exchange Server, administrators must add at least one Microsoft Exchange Server class to NetBackup that defines the appropriate schedules for that class. Administrators also must configure NetBackup to perform backup and restore operations for individual mailboxes and folders.

Most requirements for Microsoft Exchange Server classes are the same as requirements for

Although administrators can back up all databases individually, backing up an entire storage group at one time is recommended.

file system backups. Refer to the *NetBackup 3.4 System Administrator's Guide for Windows NT/2000* for detailed configuration instructions.

NetBackup for Exchange features

The following section highlights some features of NetBackup for Exchange.

Online backup. Administrators do not need to take the Microsoft Exchange Server offline while backing up the Microsoft Exchange Server data and transaction logs. This capability ensures the availability of Microsoft Exchange services and data during the Microsoft Exchange Server backup.

Reduced backup time. Administrators can perform full or incremental backups (differential incremental backup or cumulative incremental backup). A full backup may require considerable time, so administrators may perform it infrequently. In the interim, updates since the full backup can be backed up quickly and incrementally by backing up only the transaction logs. If a failure occurs, the full and incremental backups would be restored.

During recovery, the Microsoft Exchange Server will update the databases, applying each of the logged transactions to the database. After the Microsoft Exchange Server recovery has completed, the system will have been brought back to the state as it existed when the last incremental backup was performed.

Microsoft Exchange Server backup support. NetBackup supports all Microsoft Exchange Server backup methods: full backup, cumulative-incremental backup, differential-incremental backup, and copy.

Central administration. Administrators can define, back up, and restore Microsoft Exchange Servers and other NetBackup client machines from a central location.

Media management. Microsoft Exchange Server backups are saved directly to a wide variety of storage devices supported by the NetBackup master server.

Automated backups. Administrators can establish schedules for automatic, unattended backups for local or remote clients across the network. These backups can be full or incremental and are managed entirely by the NetBackup server from a central location. The administrator can also manually back up clients. To ensure consistent and accurate backups, always check database consistency before backing up a database.

Restore operations. Using a few simple operations, an administrator using the NetBackup client can browse Microsoft Exchange Server backups and select those to be restored.

In addition to backing up the Exchange databases, administrators should back up all mission-critical data such as the contents of users' mailboxes and the configuration data necessary to run the Exchange servers.

Individual mailbox backup and restore.


Administrators can perform backup and restore operations on individual mailboxes and folders. This feature includes the following capabilities:

- ▶ Scheduled backups of individual mailboxes and folders
- ▶ User-directed backups of individual mailboxes and folders
- ▶ Server- or client-based restores of individual mailboxes, folders, or messages
- ▶ Alternate mailbox and folder restores
- ▶ Alternate client restores of individual mailboxes, folders, or messages
- ▶ Software compression of backups
- ▶ Multiple data streams for backups

The Exchange Messaging API (MAPI) allows VERITAS NetBackup for Exchange to perform “brick-level” backups of Exchange mailboxes. This capability enables easy recovery of individual mailboxes, folders, or e-mail messages. Administrators no longer need to rely on a spare server to restore individual messages from Exchange.

Simplifying data protection

Understanding backup and restore processes and the interaction between system elements allows administrators to implement effective contingency plans for outages or disasters. Before implementing any backup solution in a production environment, test all proposed backup solutions in a test lab environment that matches the existing production environment.

Tools such as VERITAS NetBackup for Exchange simplify regular backups, help companies to restore important data, and minimize disruption in Exchange environments. VERITAS delivers freedom of choice and simplicity of management, providing software that ensures continuous availability of data across multiple applications, servers, storage, or networks. 

VERITAS Software Corporation is a leading provider of data availability software solutions that enable customers to protect and access their business-critical data.

FOR MORE INFORMATION

For more information, service, and support, please visit <http://www.veritas.com>
<http://seer.support.veritas.com>
or call 1-800-729-7894, reference code 8610113AW