

NDMP

Network Data Management Protocol

Using the NDMP

Workflow Analysis

Protocol Version: 2.0
Document Version: 1.2
Last Update: 07/20/98 9:35 AM
Printed: 07/20/98 9:35 AM

1. OVERVIEW.....	3
1.1 MOTIVATION	3
1.2 AUDIENCE.....	3
1.3 DOCUMENT ORGANIZATION	3
2. SESSION AUTHENTICATION.....	3
3. BACKUP.....	5
3.1 BACKUP EXCEPTIONS	8
3.1.1 End-of-media	8
3.1.2 Media errors.....	9
3.1.3 User aborted.....	10
3.1.4 Loss of data connection.....	10
4. RESTORES.....	11
4.1 RESTORE.....	11
4.2 RESTORE EXCEPTIONS.....	13
4.2.1 End-of-file	13
4.2.2 Media error	14
4.2.3 User aborted.....	15
4.2.4 Direct access restores	15
4.2.5 Loss of data connection.....	16
5. USING A JUKEBOX.....	17
5.1 BACKING UP AND RESTORING USING A JUKEBOX	17
5.2 INITIALIZING A JUKEBOX	18
5.3 EXCEPTION HANDLING.....	18
6. BROKEN CONNECTION.....	18
FIGURE 1 - MD5 AUTHENTICATION PROCESS	5

1. Overview

1.1 Motivation

This document will serve to provide a context for an implementer of an NDMP server. It is not necessary to understand how the protocol is used to be able to implement a server. It should be possible to implement a server by using the protocol document alone. However, this document allows an implementer to understand how the protocol was intended to be used and the typical sequencing of messages to be expected.

While developing the NDMP this document will be used to provide a means of design verification by fully exploring the use of the NDMP. The workflow herein describes session authentication as well as various backup/restore scenarios.

1.2 Audience

This document is intended for use by software developers of NDMP server implementations. The reader is assumed to be familiar with NDMP specifications and with the general operation of backup applications. The user is not expected to have knowledge of internal backup application behavior.

1.3 Document organization

This document will discuss the various workflow scenarios at a very high level. Message exchanges may be described by referring to a group of messages (an “interface”) or by referring to a specific message. This discussion will attempt to describe the purpose of each exchange, without the confusion of excessive detail.

Backup and restore workflow will be described for a 3 way backup. This means that the backup management software (NDMP client) is on one host, the data to be backed up is on a second host and the tape drive is on a third host. Local backups (data and tape drive on same host) or remote backups (control and tape drive on same host) are simplifications of the 3 way backup. The only changes for these simpler configurations is that a second connection may not be required.

2. Session Authentication

This section describes the three types of session authentication that are present in NDMP.

Before anything else the NDMP client needs to open a communication channel to an NDMP server, by following these steps:

- a) If available, use the `inetd` server to connect to a port that provides the NDMP service. If `inetd` is not present or the service is not known, then connect to a predefined port number (currently 10,000).
- b) A successful connection to the port will create a new server state. This may be implemented in different ways but represents a state that this connection alone controls. This state is referred to as the NDMP server. The NDMP server will also send an `NDMP_NOTIFY_CONNECTED` message back to the NDMP client to indicate a successful connection.
- c) If the connection fails for any reason, the NDMP server will send an `NDMP_NOTIFY_CONNECTED` message back to the NDMP client to indicate an error.
- d) The NDMP client will send `NDMP_CONNECT_OPEN` message to negotiate a compatible protocol implementation version. If a compatible version is not available then the NDMP client will send an `NDMP_CONNECT_CLOSE` message to close the connection.

An NDMP server only needs to support one of the three possible methods. To determine what

authentication methods the NDMP server that you are connected to supports you can issue an *NDMP_CONFIG_GET_HOST_INFO* message. The supported methods will be returned in the reply message and can be one of:

NONE

- a) No authentication is required. However, the NDMP client still needs to send an *NDMP_CONNECT_CLIENT_AUTH* message to inform the NDMP server that it wishes to continue with the connection in a non-authenticated state.

TEXT

- a) The connection is authenticated by issuing an *NDMP_CONNECT_CLIENT_AUTH* message containing a user name and clear text (non-encrypted) password.

MD5

- a) The NDMP client sends an *NDMP_CONFIG_GET_AUTH_ATTR* message so that the NDMP server can provide a session challenge. This is a 512 bit randomly generated per session quantity.
- b) The NDMP client concatenates the client password followed by some amount of null padding, the server challenge and finally a second copy of the client password. This is passed through the MD5 algorithm to generate a 128 bit client digest which is sent along with the auth id within an *NDMP_CONNECT_CLIENT_AUTH* message request.
- c) The NDMP server then creates a 128-bit MD5 digest duplicate by concatenating its copy of the password - the “shared secret” - followed by some amount of null padding, the server challenge and a second copy of the server password. It then compares its value to the value passed to it by the NDMP client. If it matches then it has authenticated the NDMP client, otherwise it refuses to authenticate the connection.
- d) The NDMP server has not been authenticated by the NDMP client at this stage, so to prevent a “rogue” NDMP server spoofing an NDMP client. The NDMP client can *optionally* prepare a client challenge (another randomly generated 512 bit value) and send it within an *NDMP_CONNECT_SERVER_AUTH* message request.
- e) The NDMP server then creates a server digest by concatenating the server password followed by some amount of null padding, the client challenge and a second copy of the server password. Finally passing this through the MD5 algorithm to create a 128 bit server digest. The digest is sent back in the *NDMP_CONNECT_SERVER_AUTH* message reply with the auth id.
- f) Finally the NDMP client compares the NDMP server’s digest against another that it has generated by concatenating the client password followed by some amount of null padding, the client challenge and a second copy of the client password passed through the MD5 algorithm. If they match then the client has authenticated the server and the connection can be trusted. If they don’t match the NDMP client should immediately close the connection with an *NDMP_CONNECT_CLOSE* message.

The amount of null padding used in generating any of the digests is determined by the length of the password (shared secret). The equation to determine the amount of null padding is:

$$\text{amount of null padding} = 128 \text{ bytes} - \text{sizeof(Challenge)} - (2 * \text{sizeof>Password});$$

Note: The client password and server password described above are the same value. The client password is the copy held by the client and the server password is the copy held by the server.

The MD5 authentication is depicted in figure 1.

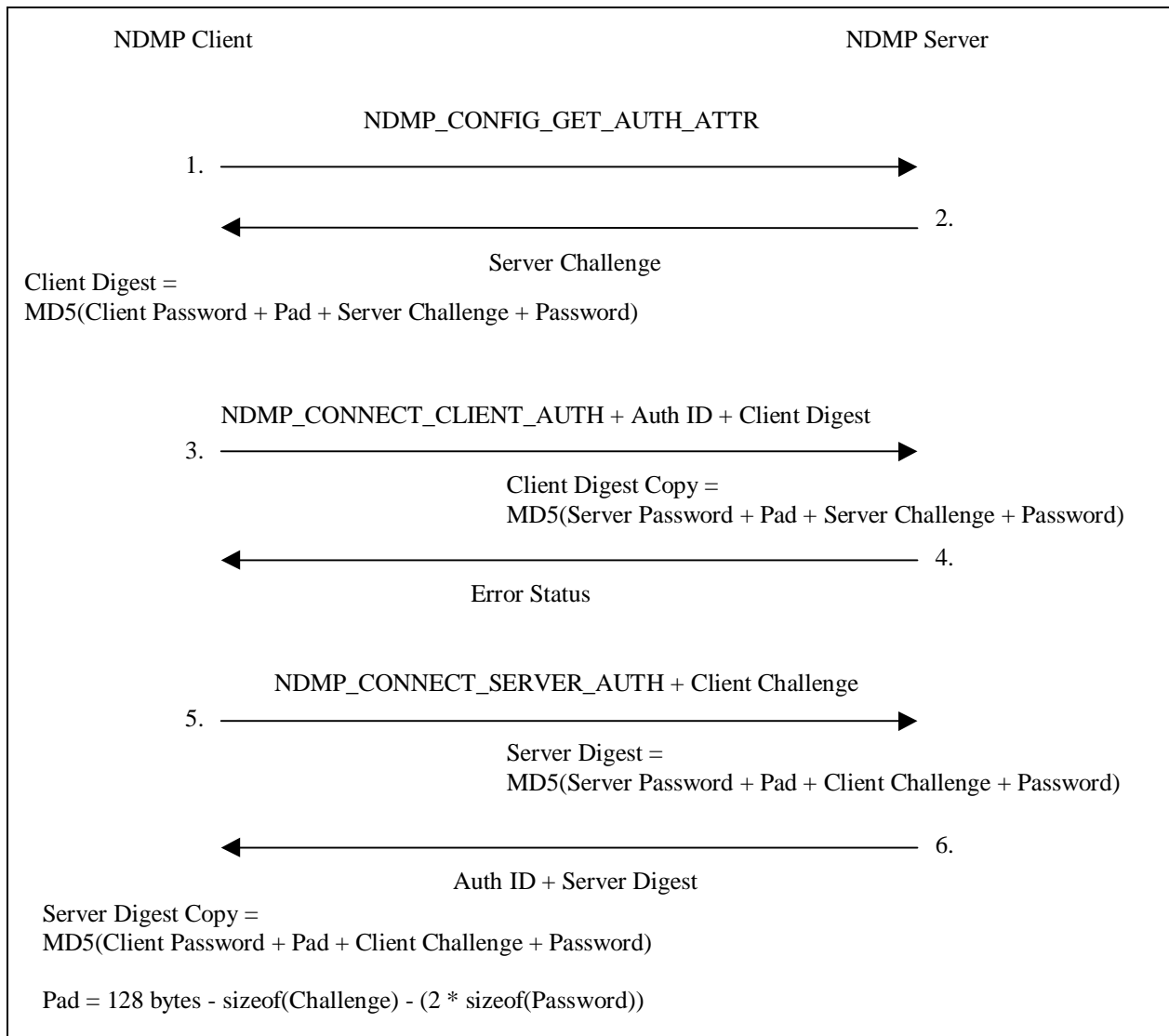


Figure 1 - MD5 authentication process.

3. Backup

This section describes the control sequence and the data flow of a backup. It assumes that the tape drive to be used is attached to the host running the NDMP TAPE server and that the data to be backed up is on the host running the NDMP DATA server.

1. NDMP client will open a communication channel to the NDMP TAPE server, negotiate the protocol version to be used and authenticate the connection.
2. Prepare the tape in the drive for backup.
 - a) The NDMP client will send an *NDMP_TAPE_OPEN* message to the NDMP TAPE server to instruct it to open a tape drive for reading. The open should be exclusive and should prevent other users from accessing the tape drive.
 - b) The NDMP client will send an *NDMP_TAPE_READ* message to read the volume label from the tape. If the volume label is invalid or the tape cannot be used for backup, then the NDMP client will send an *NDMP_TAPE_MTIO* message to rewind and eject the tape

- and the NDMP client will load a new tape (manually or using the jukebox).
- c) The NDMP client will use the TAPE interface messages to instruct the NDMP TAPE server to properly position the tape for writing the backup data. This may include
 - i) Rewinding and writing a new label,
 - ii) Forward spacing the tape and reading the last trailer file
 - d) The NDMP client will send an *NDMP_MOVER_SET_RECORD_SIZE* message to tell the NDMP TAPE server what record size to use when writing to the tape.
3. Connection to the NDMP DATA server
- a) If the data is on the same host as the tape drive, then the NDMP client is not required to open a second connection. In this case, the following references to NDMP DATA server can be replaced by NDMP TAPE server and the remainder of this step (3) should be skipped.
 - a) If the data to be backed up is on a different host than the tape drive, then the NDMP client will open an NDMP connection to the new host. This new host will be referred to as the NDMP DATA server.
4. The NDMP client will send *NDMP_CONFIG_GET_BUTYPE_ATTR* message to query the capability of NDMP DATA server. (For example, is file history is supported or not?)
5. Get a mover address from the NDMP TAPE server
- a) The NDMP client will send *NDMP_CONFIG_GET_MOVER_TYPE* message to the NDMP DATA server and NDMP TAPE server to query the type of the connections supported .
 - b) The NDMP client will choose the type of connection method which will be used between the NDMP DATA server and the NDMP TAPE server and include it in the *NDMP_MOVER_LISTEN* message.
 - c) The NDMP client will send an *NDMP_MOVER_LISTEN* message to the NDMP TAPE server.
 - d) The mover on the host running the NDMP TAPE server will create a connection point and begin listening for a connection. The NDMP TAPE server will return the mover address.
6. The NDMP client will initiate a backup.
- a) The NDMP client will use the NDMP TAPE interface of the NDMP TAPE server to write a header file followed by a file mark.
 - b) The NDMP client will send an *NDMP_DATA_START_BACKUP* message to the NDMP DATA server to begin the backup. This message will include the address of the mover on the NDMP TAPE server. The NDMP DATA server will then attempt to open a data connection to the mover on the NDMP TAPE server. If the NDMP DATA server cannot connect to the mover, then it will return an error and the NDMP client will send an *NDMP_MOVER_ABORT* message to the mover on the NDMP TAPE server to tell it to stop listening for a connection.
 - c) The *NDMP_DATA_START_BACKUP* message will define what should be backed up and what type of backup to perform. The NDMP client may include parameters that will modify the behavior of the backup. Parameters could be the dump level, compression or encryption flags, etc.
 - d) The NDMP DATA server will begin to generate data and send it to the mover.
 - e) The mover will buffer the data into tape records and write the data to tape.
7. As the backup is running, the NDMP client will be prepared to accept various messages from the NDMP DATA server and the NDMP TAPE server

- a) As individual files are backed up, the NDMP DATA server may send *NDMP_FH_ADD_UNIX_PATH* messages to the NDMP client.
- b) Both the NDMP TAPE server and the NDMP DATA server may send *NDMP_LOG_LOG* or *NDMP_LOG_DEBUG* messages to the NDMP client to indicate progress.
- c) If an event occurs that requires attention, the NDMP DATA server or NDMP TAPE server will use the NDMP NOTIFY interface to let the NDMP client know that attention is required.

8. Successful backup completion

- a) On completion of a successful backup the NDMP DATA server will close the connection to the mover and then send an *NDMP_NOTIFY_DATA_HALTED* message with *NDMP_DATA_HALT_SUCCESSFUL* reason to the NDMP client.
- b) The NDMP client will issue an *NDMP_DATA_GET_STATE* and *NDMP_DATA_GET_ENV* to the NDMP DATA server and save any prudent information returned for use during the restore process.
- c) The NDMP client will send an *NDMP_DATA_STOP* message to the NDMP DATA server.
- d) Once the NDMP DATA server has release the resources, it will return the status to the NDMP client.
- e) Since the mover on the NDMP TAPE server detects the disconnection from the NDMP DATA server, it will null pad the last tape record and then send an *NDMP_NOTIFY_MOVER_HALTED* message with *NDMP_MOVER_CONNECT_CLOSED* reason to the NDMP client.
- f) The NDMP client will issue an *NDMP_MOVER_GET_STATE* message to the NDMP TAPE server and note the total number of bytes generated.
- g) The NDMP client will send an *NDMP_MOVER_STOP* message to the NDMP TAPE server.
- h) The NDMP client will use the NDMP TAPE interface on the NDMP TAPE server to write a file mark to tape.
- i) The NDMP client will then use the NDMP TAPE interface to write a trailer file and another file mark.

9. If the NDMP client has more backup requests to process

- a) If the data to be backed up is on another host, then the NDMP client will send an *NDMP_CONNECT_CLOSE* message to close the connection to the NDMP DATA server and then open a new connection with the new NDMP DATA server.
- b) The NDMP client will use the NDMP TAPE interface to write another header file and file mark.
- c) The NDMP client will get another mover address as in step 5.
- d) The NDMP client will initiate another backup as in step 6.

10. If the NDMP client has no more backups to process

- a) The NDMP client will send an *NDMP_CONNECT_CLOSE* message to close the connection to the NDMP DATA server, unless the NDMP DATA server and NDMP TAPE server are running on the same host.
- b) The NDMP client will use the NDMP TAPE interface on the NDMP TAPE server to rewind the tape.
- c) The NDMP client may choose to use the NDMP TAPE interface to eject the tape.
- d) The NDMP client will send *NDMP_CONNECT_CLOSE* message to close the connection

to the NDMP TAPE server.

3.1 Backup Exceptions

The previous workflow assumes that there were no problems writing to tape and that everything fits on a single tape. In this section we examine some of the exceptions that can occur and how they are handled.

3.1.1 End-of-media

If the amount of data to be backed up is greater than the space available on tape, then the mover on the NDMP TAPE server will detect an end-of-media (EOM) condition before the backup is completed. This section describes how the EOM should be handled.

Detecting an end-of-media condition

- a) The mover on NDMP TAPE server detects that not all of the data was successfully written to tape. This is usually indicated as a partial write by the device driver to the tape.
 - b) The mover will update the amount of data successfully written and will change its mover state to *NDMP_MOVER_STATE_PAUSED* and the *mover_pause_reason* to *NDMP_MOVER_PAUSE_EOM*. The unwritten data will be saved for writing at a later time.
 - c) The NDMP TAPE server will send an *NDMP_NOTIFY_MOVER_PAUSED* message with an *NDMP_MOVER_PAUSE_EOM* reason to the NDMP client.
 - d) The NDMP client will query the NDMP mover_state and will remember the amount of data written to tape.
2. If the user has specified that backups may not span multiple tapes
- a) An *NDMP_DATA_ABORT* message is sent to the NDMP DATA server.
 - b) The NDMP DATA server will discard any unwritten data and close the connection to the mover on the NDMP TAPE server
 - c) The NDMP DATA server will change the data status to *NDMP_DATA_STATE_HALTED* and the reason to *NDMP_DATA_HALT_ABORTED* and then send an *NDMP_NOTIFY_DATA_HALTED* message with an *NDMP_DATA_HALT_ABORTED* reason to the NDMP client.
 - d) The NDMP client will send an *NDMP_DATA_STOP* message to the NDMP DATA server.
 - e) Once the resources have been released the NDMP DATA server will return the status to the NDMP client.
 - f) The NDMP client will receive an *NDMP_NOTIFY_MOVER_HALTED* message with *NDMP_MOVER_CONNECT_CLOSED* reason from the mover on the NDMP TAPE server.
 - a) The NDMP client will send an *NDMP_MOVER_STOP* message to the NDMP TAPE server.
3. Unloading the tape
- a) The NDMP client will use the NDMP TAPE interface to attempt to write a file mark on the tape.
 - b) The NDMP client will use the NDMP TAPE interface to rewind and eject the tape.
 - c) The NDMP client will use the NDMP TAPE interface to close the tape device.
4. Loading a new volume
- a) The NDMP client will load another tape into a drive (manually or using the jukebox)
 - b) The NDMP client will use the NDMP TAPE interface to open the new tape device.

- c) The NDMP client will use the NDMP TAPE interface to prepare the tape for the backup data in the same fashion as in the local backup.
5. Continuing the backup
- a) If the backup is not allowed to span multiple tapes, then the backup is restarted as in step 5 and 6 of the backup workflow.
 - b) If the backup is not restarted, then the NDMP client will send an *NDMP_MOVER_CONTINUE* message to the NDMP TAPE server.
 - c) The mover on the NDMP TAPE server will combine the data that was not written to tape with new backup data to create a full sized tape record.
 - d) The full size record is written to tape.
 - e) The backup continues.

3.1.2 Media errors

Many tape drives have read-after-write capability and can detect write errors. This section describes how the media error should be handled.

1. Detecting a media error

- a) The mover on the NDMP TAPE server somehow detects a media error. This is usually detected by the tape drive and returned by the device driver.
- b) The NDMP TAPE server will change its mover state to *NDMP_MOVER_STATE_PAUSED* and the reason to *NDMP_MOVER_PAUSE_MEDIA_ERROR*.
- c) The NDMP TAPE server will send an *NDMP_NOTIFY_MOVER_PAUSED* message to the NDMP client with an *NDMP_MOVER_PAUSE_MEDIA_ERROR* reason .
- d) The NDMP client will send an *NDMP_DATA_ABORT* message to the NDMP DATA server. The NDMP DATA server will close the connection to the mover on the NDMP TAPE server and will change its state to *NDMP_DATA_STATE_HALTED* and the reason to *NDMP_DATA_HALT_ABORTED*.
- e) The NDMP client will send an *NDMP_DATA_STOP* message to the NDMP DATA server.
- f) Once the resources have been released the NDMP DATA server will return the status to the NDMP client.
- g) The NDMP client will send an *NDMP_MOVER_ABORT* message to the NDMP TAPE server .
- h) The NDMP TAPE server will change its mover state to *NDMP_MOVER_STATE_HALTED* and the reason to *NDMP_MOVER_HALT_ABORTED*.
- i) The NDMP TAPE server will send an *NDMP_NOTIFY_MOVER_HALTED* message to the NDMP client with an *NDMP_MOVER_HALT_ABORTED* reason .
- j) The NDMP client will send an *NDMP_MOVER_STOP* message to the mover on the NDMP TAPE sever.

2. Handling the Media error

- a) The NDMP client host will use the NDMP TAPE interface to rewind and eject the tape without writing a file mark.
- b) The NDMP client will close the tape device.

3. The NDMP client will load another volume as in the EOM workflow.

4. Restarting the backup

- a) The NDMP client will use the *NDMP_DATA_START_BACKUP* to start the backup over.

3.1.3 User aborted

It is possible for the user to abort a backup in progress. This section describes how that is handled.

1. Sending an abort.

- a) The NDMP client uses the NDMP DATA interface to send an *NDMP_DATA_ABORT* message to the NDMP DATA server.
- b) The NDMP DATA server will change the data state to *NDMP_DATA_STATE_HALTED* and the reason to *NDMP_DATA_HALT_ABORTED*. Unwritten data is discarded. No further backup data or file history will be generated.
- c) The NDMP DATA server will close the connection to the mover on the NDMP TAPE server.
- d) The NDMP DATA server will send an *NDMP_NOTIFY_DATA_HALTED* message with an *NDMP_DATA_HALT_ABORTED* reason to the NDMP client host.
- e) The NDMP client will send an *NDMP_DATA_STOP* message to the NDMP DATA server.
- f) Once the resources have been released the NDMP DATA server will return the status to the NDMP client.
- g) The NDMP client will receive an *NDMP_NOTIFY_MOVER_HALTED* message from the NDMP TAPE server with the reason set to *NDMP_MOVER_CONNECT_CLOSED*.
- h) The NDMP client will send an *NDMP_MOVER_STOP* message to the NDMP TAPE server.

2. Handling the abort

- a) The NDMP client host will use the NDMP TAPE interface on the NDMP TAPE server to write a file mark to tape.
- b) The NDMP client host will use the NDMP TAPE interface to write a trailer record that indicates that the backup was not complete, followed by a file mark.
- c) The file history collected by the NDMP client will be discarded.

3. Continuing

- a) The NDMP client may or may not continue with the next backup request.
- b) If there are no more requests, then the NDMP client will use the NDMP TAPE interface to rewind and eject the tape. The NDMP client will then send an *NDMP_CONNECT_CLOSE* message to the NDMP TAPE server to close the connection.

3.1.4 Loss of data connection

The loss of data connection can be detected from the NDMP DATA server or from the NDMP TAPE server.

1. Detected from the NDMP DATA server:

- a) The NDMP DATA server gets an error while writing to the data connection.
- b) The NDMP DATA server will change the data state to *NDMP_DATA_STATE_HALTED* and the reason to *NDMP_DATA_HALT_CONNECT_ERROR*. Unwritten data is discarded. No further backup data or file history will be generated.
- c) The NDMP DATA server will close the connection to the mover on NDMP TAPE server.

- d) The NDMP DATA server sends an *NDMP_NOTIFY_DATA_HALTED* message to the NDMP client with a reason of *NDMP_DATA_HALT_CONNECT_ERROR*.
 - e) The NDMP client will send an *NDMP_DATA_STOP* message to the NDMP DATA server.
 - f) The NDMP client will send an *NDMP_MOVER_ABORT* message to the NDMP TAPE server.
 - g) The NDMP client will receive an *NDMP_NOTIFY_MOVER_HALTED* message from the NDMP TAPE server with the reason set to *NDMP_MOVER_CONNECT_CLOSED* or *NDMP_MOVER_HALT_ABORTED* depending on the sequence to detect the disconnection from the NDMP DATA server first or receive an *NDMP_MOVER_ABORT* message.
2. Detected from the NDMP TAPE server:
- a) The NDMP TAPE server gets an error while reading from the data connection.
 - b) The NDMP TAPE server sends an *NDMP_NOTIFY_MOVER_HALTED* message with the reason set to *NDMP_MOVER_HALT_CONNECT_ERROR*.
 - c) The NDMP client will use the NDMP DATA interface to send an *NDMP_DATA_ABORT* message to the NDMP DATA server.
 - d) The NDMP DATA server will change the data state to *NDMP_DATA_STATE_HALTED* and the reason to *NDMP_DATA_HALT_ABORTED*. Unwritten data is discarded. No further backup data or file history will be generated.
 - e) The NDMP DATA server will close the connection to the mover on NDMP TAPE server.
 - f) The NDMP DATA server will send an *NDMP_NOTIFY_DATA_HALTED* message with an *NDMP_DATA_HALT_ABORTED* reason to the NDMP client.
 - g) The NDMP client will send an *NDMP_DATA_STOP* message to the NDMP DATA server.
 - h) Once the resources have been released the NDMP DATA server will return the status to the NDMP client.
 - i) The NDMP client will send an *NDMP_MOVER_STOP* message to the NDMP TAPE server.

4. Restores

4.1 Restore

This section describes the restore process. It assumes that the data to be restored is already in a tape drive.

1. NDMP client will open a communication channel to the NDMP TAPE server, negotiate the protocol version to be used and authenticate the connection.
2. Prepare the tape in the drive for restore.
 - a) The NDMP client will send an *NDMP_TAPE_OPEN* message to the NDMP TAPE server to instruct it to open a tape drive for reading. The open should be exclusive and should prevent other users from accessing the tape drive.
 - b) The NDMP client will send an *NDMP_TAPE_READ* message to read the volume label from the tape. If the volume label is invalid then the NDMP client will send an *NDMP_TAPE_MTIO* message to rewind and eject the tape.
3. Positioning the tape
 - a) The NDMP client will use the NDMP TAPE interface to position to the header file for

- the backup data that is to be restored.
- b) The NDMP client will use the NDMP TAPE interface to read the header file.
 - c) If the header is incorrect or cannot be read, the NDMP client will rewind and eject the tape.
 - d) The NDMP client will use the NDMP TAPE interface to position past the file mark.
 - e) The NDMP client will send an *NDMP_MOVER_SET_WINDOW* message to identify the begin and end bytes of the current tape file relative to the backup data.
4. If the data is to be restored to a machine that is different than the machine that has the tape drive attached, then the NDMP client will open a second NDMP connection to the host with the data.
 5. The NDMP client will send an *NDMP_CONFIG_GET_BUTYPE_ATTR* message to query the capability of the NDMP recover utility on the host running the NDMP DATA server. (For example, is individual file restore supported or not?)
 6. Begin restore process on the NDMP DATA server
 - a) The NDMP client will send an *NDMP_CONFIG_GET_MOVER_TYPE* message to NDMP DATA server and NDMP TAPE server to query the type of the connection supported .
 - b) The NDMP client will choose the type of connection method which will be used between the NDMP DATA server and the NDMP TAPE server and include it in the *NDMP_MOVER_LISTEN* message.
 - c) The NDMP client will send an *NDMP_MOVER_LISTEN* message to the NDMP TAPE server. The mover will return the address on which it will begin to listen.
 - d) The NDMP client will send an *NDMP_DATA_START_RESTORE* message to the NDMP DATA server. The message will include the mover address, the list of files to be restored and the destination.
 - e) If the restore involves a remote NDMP server, i.e. not a local retrieval
 - i) The NDMP DATA server will send an *NDMP_NOTIFY_DATA_READ* message to the NDMP client to initiate the restore.
 - ii) The NDMP client should send an *NDMP_MOVER_READ* message to the NDMP TAPE server to inform the mover to start sending the requested data.
 - f) The NDMP client will be prepared to accept NDMP LOG messages.
 7. Reading the last tape record
 - a) When the last tape record is read, the pad bytes are discarded.
 - b) If the file mark is consumed, then the *tape_status* must reflect that fact.
 8. Successful restore
 - a) The NDMP DATA server will send an *NDMP_LOG_FILE* message to report if the files are restored.
 - b) Once all of the files have been recovered, the NDMP DATA server will change its status to *NDMP_DATA_STATE_HALTED* and the reason to *NDMP_DATA_HALT_SUCCESSFUL*. It will close the connection to the mover on the NDMP TAPE server and send an *NDMP_NOTIFY_DATA_HALTED* message to the NDMP client.
 - c) The NDMP client will send an *NDMP_DATA_STOP* message to the NDMP DATA server.
 - d) Once the resources have been released the NDMP DATA server will return the status to the NDMP client.
 - e) The NDMP client will receive an *NDMP_NOTIFY_MOVER_HALTED* message with an

NDMP_MOVER_CONNECT_CLOSED reason from the NDMP TAPE server

- f) The NDMP client will send an *NDMP_MOVER_STOP* message to the NDMP TAPE server.
- g) If there are more restores to be processed from the tape, the NDMP client will position the tape as above.
- h) If there are no more restores to be processed for this tape, the NDMP client will use the NDMP TAPE interface to rewind and eject the tape.
- i) The NDMP client will close the tape device.
- j) The NDMP client will close the connection.

4.2 Restore Exceptions

4.2.1 End-of-file

If the NDMP client can support backups that span multiple tape files, then during a restore, it is possible to reach an end-of-file mark before all of the data to be restore has been read. This section describes how that condition should be handled.

1. Detect end-of-file

- a) The mover on the NDMP TAPE server detects an end-of-file condition. This is normally detected by the tape drive and returned as a partial read by the device driver.
- b) The mover processes the data that was actually read.
- c) The NDMP TAPE server changes the mover status to *NDMP_MOVER_STATE_PAUSED* and the reason to *NDMP_MOVER_PAUSE_EOF* and sends an *NDMP_NOTIFY_MOVER_PAUSED* message with an *NDMP_MOVER_PAUSE_EOF* reason to the NDMP client.

2. More tape files associated with the backup image:

- a) If the NDMP client needs to select another tape.
 - i) The NDMP client will use the NDMP TAPE interface to rewind and eject the tape.
 - ii) The NDMP client will use the NDMP TAPE interface to close the tape drive and open another.
 - iii) The NDMP client will use the NDMP TAPE interface to verify the volume label.
- b) The NDMP client will use the NDMP TAPE interface to position to the correct tape file.
- c) The NDMP client will use the NDMP MOVER interface to set the new mover_window.
- d) The NDMP client will send an *NDMP_MOVER_CONTINUE* message to the NDMP.
- e) The mover will continue reading data and sending it to the NDMP DATA server.
- f) If a continuation tape cannot be located, then the NDMP client will send an *NDMP_DATA_ABORT* message to the NDMP DATA server and the restore will be aborted. The NDMP DATA server will change state to halted and will send an *NDMP_NOTIFY_DATA_HALTED* message to the NDMP client with an *NDMP_DATA_HALT_ABORTED* reason. The NDMP DATA server will close the data connection to the NDMP TAPE server.
- g) The NDMP client will send an *NDMP_DATA_STOP* message to return the NDMP DATA server to an idle state.
- h) Once the resources have been released the NDMP DATA server will return the status to

the NDMP client.

- i) The NDMP TAPE server will detect the closed data connection and change it's mover state to *NDMP_MOVER_STATE_HALTED*. The NDMP TAPE server will send an *NDMP_NOTIFY_MOVER_HALTED* with an *NDMP_MOVER_CONNECT_CLOSED* reason message to the NDMP client.
 - j) The NDMP client will send an *NDMP_MOVER_STOP* message to the NDMP TAPE server.
3. No more tape file:
- a) The NDMP client will send an *NDMP_MOVER_CLOSE* message to the NDMP TAPE server.
 - b) The mover will close the connection to the NDMP DATA server.
 - c) The mover will change it's mover state to *NDMP_MOVER_STATE_HALTED*. The NDMP TAPE server will send an *NDMP_NOTIFY_MOVER_HALTED* message with an *NDMP_MOVER_CONNECT_CLOSED* reason message to the NDMP client.
 - d) The NDMP DATA server will detect the end of data connection and change state to halted and will send an *NDMP_NOTIFY_DATA_HALTED* message to the NDMP client with an *NDMP_HALT_SUCCESSFUL* reason if receive the expected data, or an *NDMP_HALT_CONNECT_ERROR* reason if not receiving the expected end of data.
 - e) The NDMP DATA server will send *NDMP_LOG_FILE* message to report if the files are restored.
 - f) The NDMP client will send an *NDMP_DATA_STOP* message to return the NDMP DATA server to an idle state.
 - g) Once the resources have been released the NDMP DATA server will return the status to the NDMP client.
 - h) The NDMP client will send an *NDMP_MOVER_STOP* message to the NDMP TAPE server.

4.2.2 Media error

It is possible for the tape drive to detect a media error while reading.

1. Detecting a media error

- a) The NDMP TAPE server somehow detects a media error. This is usually detected by the tape drive and returned by the device driver.
- b) The NDMP TAPE server will change its mover status to *NDMP_MOVER_STATE_PAUSED* and the mover_paused_reason to *NDMP_MOVER_PAUSE_MEDIA_ERROR*. No further processing of data will occur.
- c) The NDMP TAPE server will send an *NDMP_NOTIFY_MOVER_PAUSED* message with an *NDMP_MOVER_PAUSE_MEDIA_ERROR* reason to the NDMP client.
- d) The NDMP client will send an *NDMP_DATA_ABORT* message to the NDMP DATA server. The NDMP DATA server will close the connection to the mover on the NDMP TAPE server and will change its state to *NDMP_DATA_STATE_HALTED* and the reason to *NDMP_DATA_HALT_ABORTED*.
- e) The NDMP client will send an *NDMP_DATA_STOP* message to the NDMP DATA server.
- f) Once the resources have been released the NDMP DATA server will return the status to the NDMP client.
- g) The NDMP client will send an *NDMP_MOVER_ABORT* message to NDMP TAPE server .

- h) The NDMP TAPE server will change its mover state to *NDMP_MOVER_STATE_HALTED* and the reason to *NDMP_MOVER_HALT_ABORTED*.
 - i) The NDMP TAPE server will send an *NDMP_NOTIFY_MOVER_HALTED* message to the NDMP client with an *NDMP_MOVER_HALT_ABORTED* reason.
 - j) The NDMP client will send an *NDMP_MOVER_STOP* message to the mover on the NDMP sever.
2. Handling the Media error
- a) The NDMP client host will use the NDMP TAPE interface to rewind and eject the tape.
 - b) The NDMP client will close the tape device.

4.2.3 User aborted

It is possible for the user to abort an in progress restore. This section describes how that is handled.

1. Sending an abort.
- a) The NDMP client uses the NDMP DATA interface to send an *NDMP_DATA_ABORT* message to the NDMP DATA server.
 - b) The NDMP TAPE server will change the data state to *NDMP_DATA_STATE_HALTED* and the reason to *NDMP_DATA_HALT_ABORTED*. No further data will be processed. The connection to the mover on the NDMP TAPE server will be closed.
 - c) The NDMP TAPE server will send an *NDMP_NOTIFY_DATA_HALTED* message with an *NDMP_DATA_HALT_ABORTED* reason to the NDMP client.
2. Handling the abort
- a) The NDMP client host will send an *NDMP_DATA_STOP* message to the NDMP DATA server.
 - b) When the NDMP DATA server has released all resources it changes its state to *NDMP_DATA_STATE_IDLE* and returns the status to the NDMP client.
 - c) The NDMP client will receive an *NDMP_NOTIFY_MOVER_HALTED* message from the NDMP TAPE server with the reason set to *NDMP_MOVER_CONNECT_CLOSED*.
 - d) The NDMP client will send an *NDMP_MOVER_STOP* message to the NDMP TAPE server.
3. Continuing
- a) The NDMP client may or may not continue with the next restore request.
 - b) If there are no more requests, then the NDMP client will use the NDMP TAPE interface to rewind and eject the tape. The NDMP client will then close the connection to the NDMP TAPE server.

4.2.4 Direct access restores

The NDMP client may support a mechanism that allows the restore process to position directly to the correct tape record to perform a file restore more quickly. If the NDMP detects that tape positioning is required within the mover window, then it can perform the tape positioning without using the NDMP client, but if the tape record is outside the mover window, then the NDMP client must be used to position the tape.

1. If the data required for the restore is outside the current tape file as defined by the mover window, then the NDMP TAPE server changes the mover status to *NDMP_MOVER_STATE_PAUSED* and the reason to *NDMP_MOVER_PAUSE_SEEK* and the seek offset in the status is set to the desired offset. The NDMP sends an *NDMP_NOTIFY_MOVER_PAUSED* message with reason to *NDMP_MOVER_PAUSE_SEEK* to the NDMP client.

2. If required the NDMP client may rewind and eject the tape drive or it may close the tape device and open another device.
3. The NDMP client will position the tape to the correct tape file.
4. The NDMP client will send an *NDMP_MOVER_SET_WINDOW* message.
5. The NDMP client will use the NDMP TAPE interface to position to the tape record that contains the desired offset.
6. The NDMP client will then send an *NDMP_MOVER_CONTINUE* message to the NDMP TAPE server.
7. The NDMP TAPE server will use the current record number, the record size and the *mover_window_offset* to calculate how much of the tape record should be skipped.
8. The NDMP TAPE server will read the next tape record, skip the correct number of bytes and continue reading the data and passing it to the NDMP DATA server.

4.2.5 Loss of data connection

The loss of data connection can be detected from the NDMP DATA server or from the NDMP TAPE server.

1. Detected from the NDMP DATA server:
 - a) The NDMP DATA server gets an error while reading from the data connection.
 - b) The NDMP DATA server will change the data state to *NDMP_DATA_STATE_HALTED* and the reason to *NDMP_DATA_HALT_CONNECT_ERROR*. Unwritten data is discarded. No further backup data or file history will be generated.
 - c) The NDMP DATA server will close the connection to the mover on NDMP TAPE server.
 - d) The NDMP DATA server sends an *NDMP_NOTIFY_DATA_HALTED* message to the NDMP client with a reason of *NDMP_DATA_HALT_CONNECT_ERROR*.
 - e) The NDMP client will send an *NDMP_DATA_STOP* message to the NDMP DATA server.
 - f) The NDMP client will send an *NDMP_MOVER_ABORT* message to the NDMP TAPE server.
 - g) The NDMP client will receive an *NDMP_NOTIFY_MOVER_HALTED* message from the NDMP TAPE server with the reason set to *NDMP_MOVER_CONNECT_CLOSED* or *NDMP_MOVER_HALT_ABORTED* depending on the sequence to detect the disconnection from the NDMP DATA server first or receive an *NDMP_MOVER_ABORT* message.
2. Detected from the NDMP TAPE server:
 - a) The NDMP TAPE server gets an error while writing to the data connection.
 - b) The NDMP TAPE server sends an *NDMP_NOTIFY_MOVER_HALTED* message with the reason set to *NDMP_MOVER_HALT_CONNECT_ERROR*.
 - c) The NDMP client will use the NDMP DATA interface to send an *NDMP_DATA_ABORT* message to the NDMP DATA server.
 - d) The NDMP DATA server will change the data state to *NDMP_DATA_STATE_HALTED* and the reason to *NDMP_DATA_HALT_ABORTED*. Unwritten data is discarded. No further backup data or file history will be generated.
 - e) The NDMP DATA server will close the connection to the mover on NDMP TAPE server.
 - f) The NDMP DATA server will send an *NDMP_NOTIFY_DATA_HALTED* message with an *NDMP_HALT_ABORTED* reason to the NDMP client.

- g) The NDMP client will send an *NDMP_DATA_STOP* message to the NDMP DATA server.
- h) Once the resources have been released the NDMP DATA server will return the status to the NDMP client.
- i) The NDMP client will send an *NDMP_MOVER_STOP* message to the NDMP TAPE server.

5. Using a Jukebox

A jukebox manager application could make a connection to the NDMP server when it starts and close the connection when exiting. After the connection is established it could use the NDMP SCSI interface to open the jukebox device. This device name refers to the device that controls the mechanics of the jukebox.

5.1 *Backing up and restoring using a jukebox*

In most ways the workflow described here is identical to the previous workflow with the exception of how tapes are loaded into the drive and unloaded from the drive.

1. Loading a tape.
 - a) The jukebox manager forms and sends SCSI cdb's to determine if the jukebox inventory has changed.
 - b) The jukebox manager will determine which tape to load into what drive and form and send a SCSI MOVE MEDIUM cdb to move the tape into the drive.
 - c) The jukebox manager will open a connection to the NDMP server to which the tape drive is attached.
 - d) The jukebox manager will repeatedly attempt to open the NDMP TAPE interface to verify that the tape actually loaded and became ready.
 - e) The jukebox manager will close the connection to the NDMP server to which the tape drive is connected.
2. Unloading a tape.
 - a) The jukebox manager will form and send SCSI cdb's to determine if the jukebox inventory has changed.
 - b) The jukebox manager will form and send a SCSI MOVE MEDIUM cdb to move the tape from the tape drive to its original location. If this succeeds the jukebox manager continues as described in the local workflow sections.
 - c) The jukebox manager will cause the tape drive to unload.
 - i) The jukebox manager will open a connection to the NDMP server to which the tape drive is attached.
 - ii) The jukebox manager will use the NDMP TAPE interface to open the tape drive.
 - iii) The jukebox manager will use the NDMP TAPE interface to eject the tape drive.
 - iv) The jukebox manager will use the NDMP TAPE interface to close the tape drive.

- v) The jukebox manager will close the connection to the NDMP server to which the tape drive is attached.
- d) The jukebox manager will form and send a SCSI MOVE MEDIUM cdb to move the tape from the tape drive to its original location.

5.2 *Initializing a jukebox*

When The jukebox manager first contacts the jukebox it will form and send SCSI cdb's to determine the type and geometry of the jukebox.

1. The jukebox manager will form and send a SCSI INQUIRY cdb to obtain the product id of the jukebox.
2. The jukebox manager will form and send a SCSI MODE SENSE cdb to determine the number and physical addresses of the slots, drive and other elements of the jukebox.
3. The jukebox manager will form and send a SCSI READ ELEMENT STATUS cdb for each slot or drive to determine if the slot or tape drive is empty, full or missing.
4. The jukebox manager may form and send other SCSI cdb's depending on the product id returned by the SCSI INQUIRY.

5.3 *Exception handling*

1. If the jukebox manager detects that the jukebox inventory may have changed it will form and send a SCSI READ ELEMENT STATUS cdb for each slot or drive to determine if the slot or tape drive is empty, full or missing.
 - a) If the data returned by the SCSI READ ELEMENT STATUS cdb indicates that the jukebox is unsure of it's physical inventory, The jukebox manager will form and send a SCSI INITIALIZE ELEMENT STATUS cdb to cause the jukebox to scan its physical inventory.
 - b) If any SCSI cdb fails the jukebox manager may form and send additional SCSI cdb's to correct the problem.

6. **Broken connection**

If the TCP/IP connection between the NDMP client and the NDMP server is broken, the NDMP client will be responsible for recovery. However, the NDMP server is expected to shutdown in a manner that allows the NDMP client to reconnect.

1. NDMP server detects a broken connection
 - a) NDMP server discards any unwritten data.
 - b) NDMP server closes the tape device.
 - c) NDMP server terminates.