AFS & Kerberos Best Practices '07

An NDMP Server for AFS

Overview

Introduction to NDMP

A Brief History of NDMP

NDMP Concepts and Terminology

Example NDMP Topologies and Configurations

NDMP Specification (Interfaces, Protocol)

Current Backup and Recovery Model for AFS

Requirements for an NDMP Server for AFS

An NDMP Data Server for AFS (Interfaces)

Open Design Issues

Strengths and Weaknesses of an NDMP Solution for AFS

Introduction to NDMP

The Network Data Management Protocol (NDMP) : An open protocol for enterprise-wide, network-based data management such as backup and recovery.

Goals include interoperability, contemporary functionality, extensibility, internationalization, security, and insuring data integrity.

Data Management Application (DMA): Backup and recovery software

Data Service Providers (DSPs): A data producer and/or consumer:

- **Data service (NAS filers, AFS file servers)**
- **Tape service (tape devices)**
- **SCSI Pass-through Service (tape libraries)**

TCP/IP and the XDR record marking protocols are foundations for NDMP.

A Brief History of NDMP

1996 Dave Hitz (Network App) and Roger Stager (PDC) developed NDMP v1

Address how backup management software supports storage appliances

Data Management Application (DMA) :

- **User interface for configuring and scheduling backups**
- **Perform restores (including individual files)**

Data Service Providers (DSPs):

- **Understand the layout of and provide access to the data/storage**
- **DMA does not need to know details of file system implementation**
- **Backup and restore methods can be optimized**

A Brief History of NDMP

1996 NDMP v1

- Backup/restore using only locally attached tape devices
- Provided interfaces for file history, log messages, tape changes, etc.
- Allowed backup software to take a "hands off" approach

1996 NDMP v2

- Support for 3-way operations
- Support for remote backup & restore (dedicated backup server)
- More centralization tape library hardware

1997 NDMP v3

- Extended support for Data Service to Data Service
- Extended support for Tape Service to Tape Service operations.
- Some auto discovery, allowing a DMA to query a DSP's capabilities

A Brief History of NDMP

2003 NDMP v4 (Extensions)

- **Standard and Proprietary**
- **Documented in separate specification (no core protocol impact)**
  - **Snapshot management**
  - **Restartable Backup**
  - **File Service (disk based backups)**

NDMP v5 (wish list)

- **Multi-source - Multi-destination Sessions**
- **Improved Security**
- **Management of data objects beyond file systems**
- **Standardization of Environment Variables**

More information….

Developer web site: www.ndmp.org

Email: ndmp-tech@ndmp.org

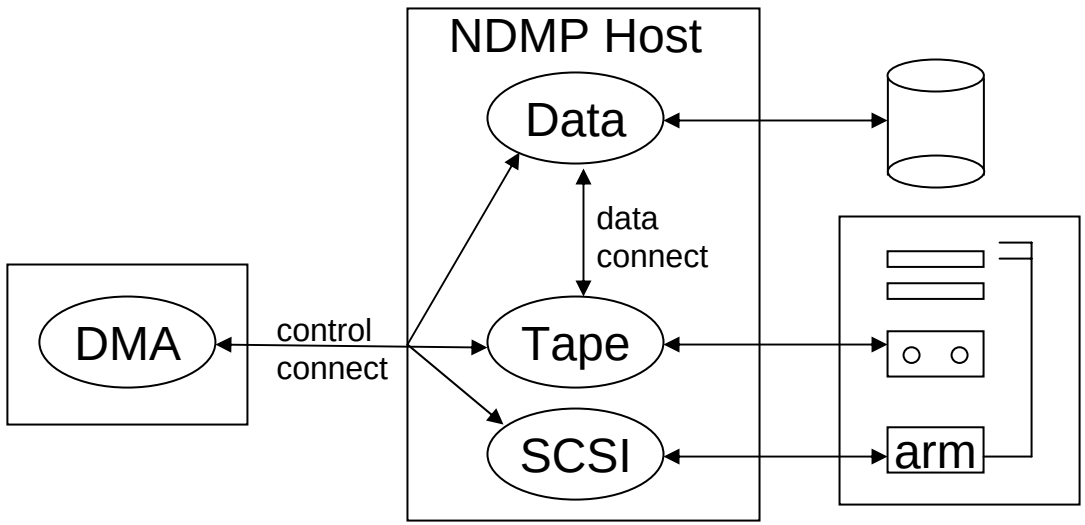
NDMP v4 specification: draft-skardal-ndmpv4-04.txt

NDMP Concepts and Terminology

Network Data Management Protocol (NDMP): Protocol used to control the NDMP services participating in a session.

Specifies format and means of transmission of messages and payload data

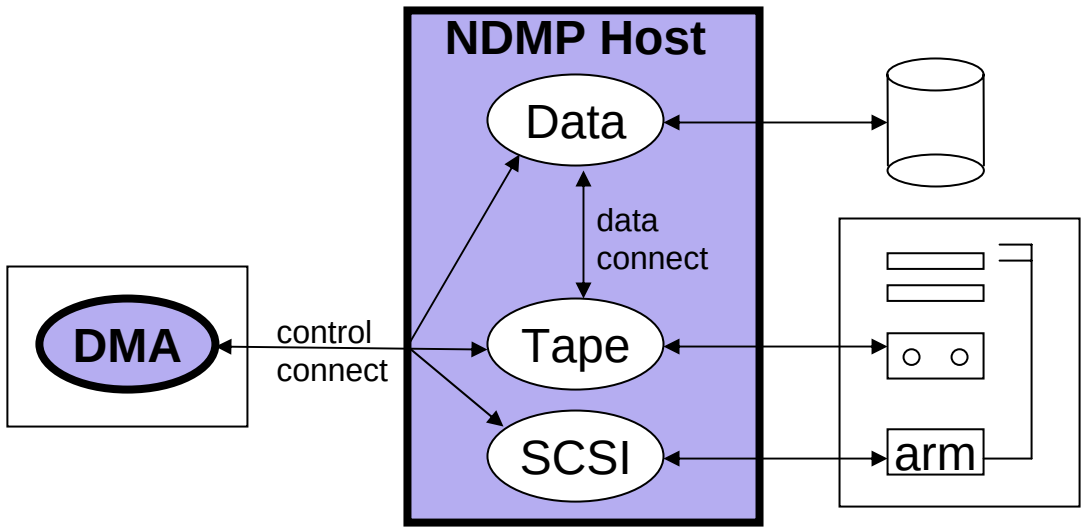
- **Between a DMA and an NDMP server**
- **Between two NDMP servers**



NDMP Concepts and Terminology

Data Management Application (DMA): Controls the NDMP session

NDMP Host: System that executes the NDMP Server application



OpenAFS

Teradactyl®
True incremental Backup System®

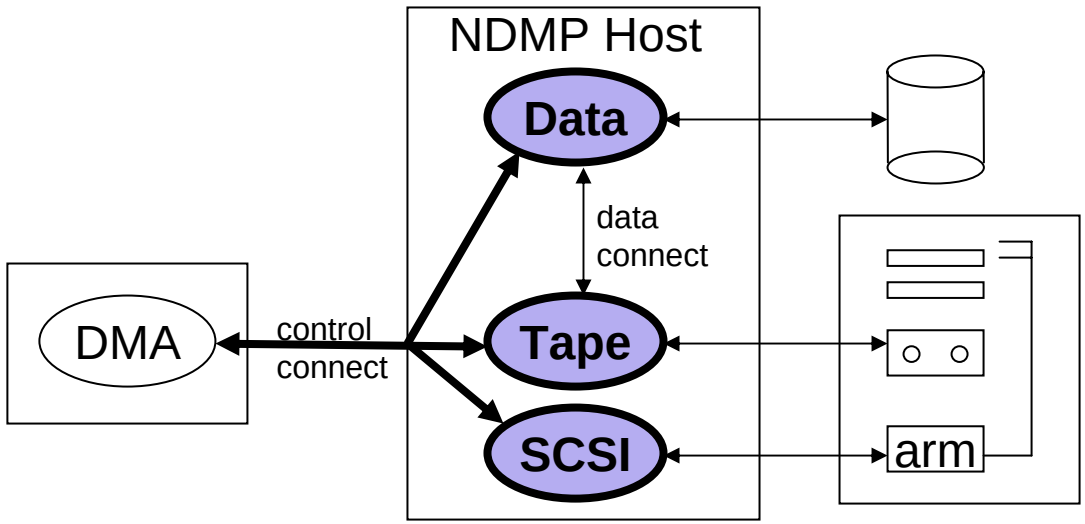
NDMP
Network Data Management Protocol

NDMP Concepts and Terminology

NDMP Service: The state machine on the NDMP Host, accessed with the Internet protocol, controlled using the NDMP protocol (Data, Tape, SCSI)

NDMP Server: An instance of one or more distinct NDMP services controlled by a single NDMP control connection

A Data/Tape/SCSI Server is an NDMP Server providing Data, Tape, and SCSI services.



OpenAFS

Teradactyl®
True incremental Backup System®

NDMP
Network Data Management Protocol

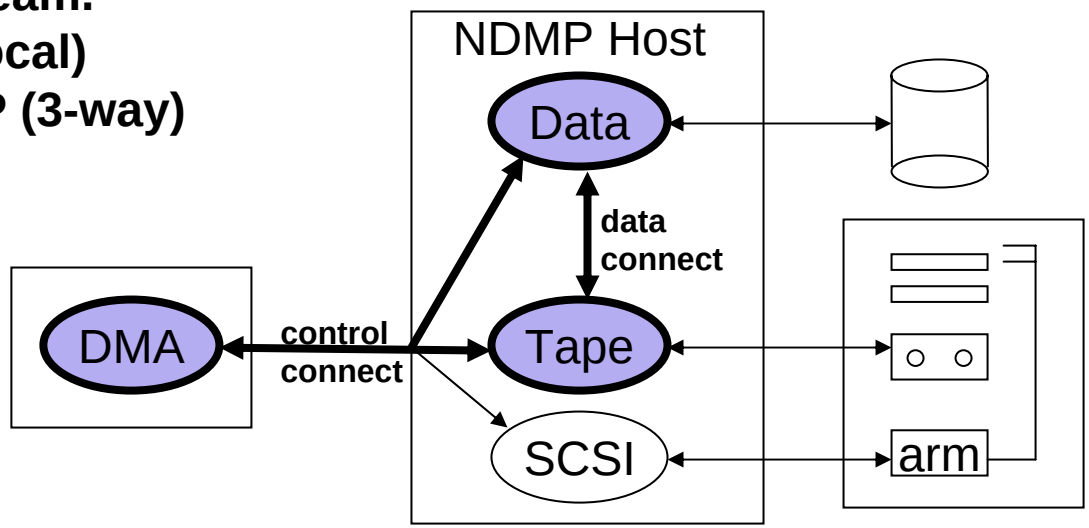
NDMP Concepts and Terminology

NDMP Session: The configuration of one DMA and two NDMP services to perform a data management operation such as a backup or a recovery.

Control Connection: A bi-directional TCP/IP connection that carries XDR encoded NDMP messages between the DMA and the NDMP Server.

Data Connection: The connection between the two NDMP Servers that carry the data stream.

- **IPC (local)**
- **TCP/IP (3-way)**

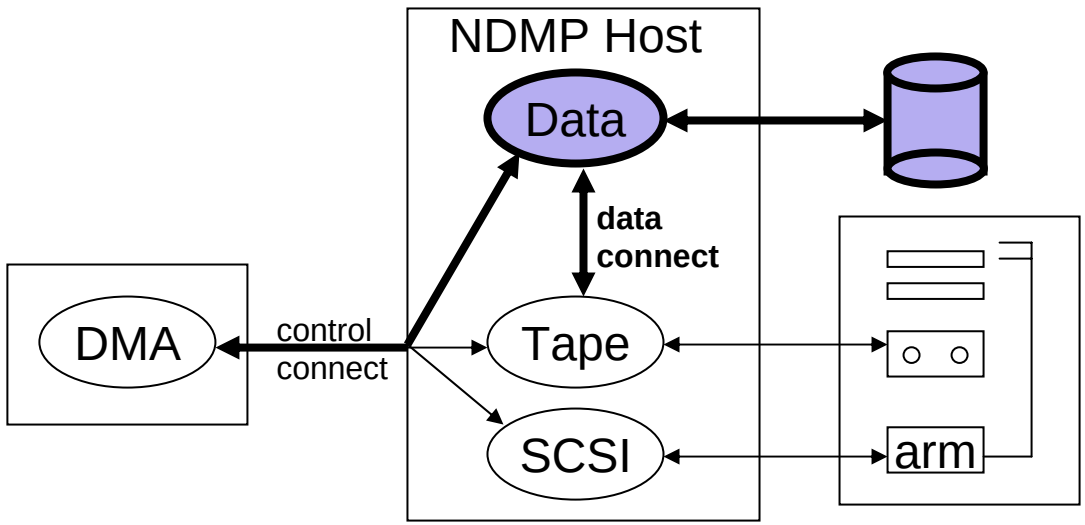


NDMP Concepts and Terminology

Data Service: A NDMP Service that transfers data between primary storage and the Data Connection.

Controlled by the DMA using Request/Reply messages

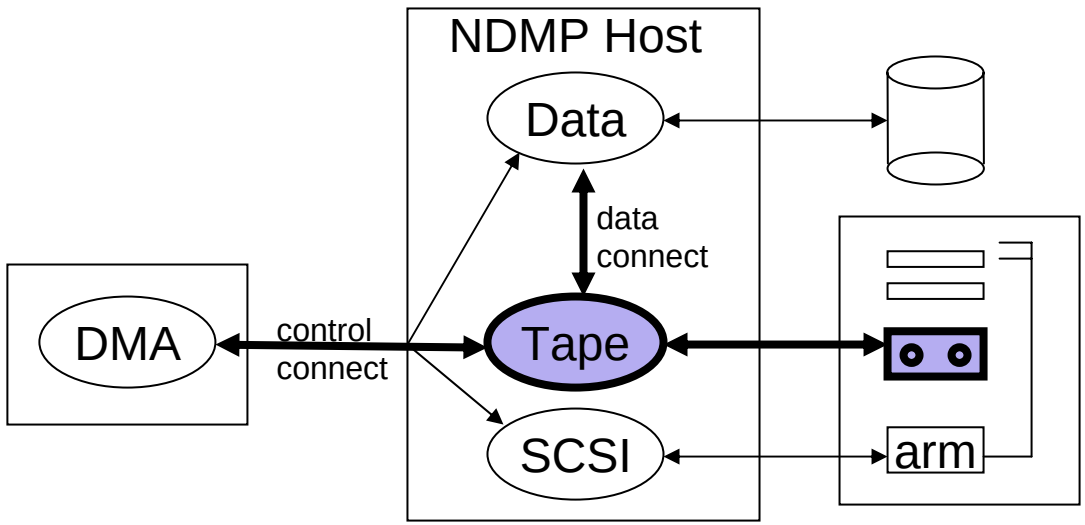
Sends Alert, Log, File History using Post messages



NDMP Concepts and Terminology

Tape Service: A NDMP Service that transfers data between secondary storage and the Data Connection and allows the DMA to manipulate and access secondary storage.

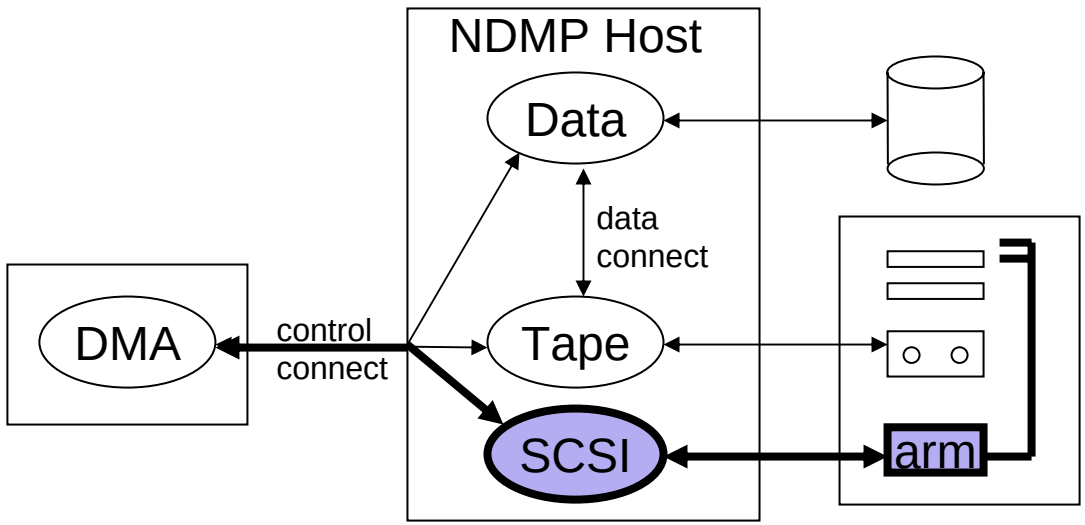
Mover: An aspect of the Tape Service that transfers data between the secondary storage and the Data Connection.



NDMP Concepts and Terminology

SCSI Service: A NDMP Service that passes low-level SCSI commands to a SCSI device

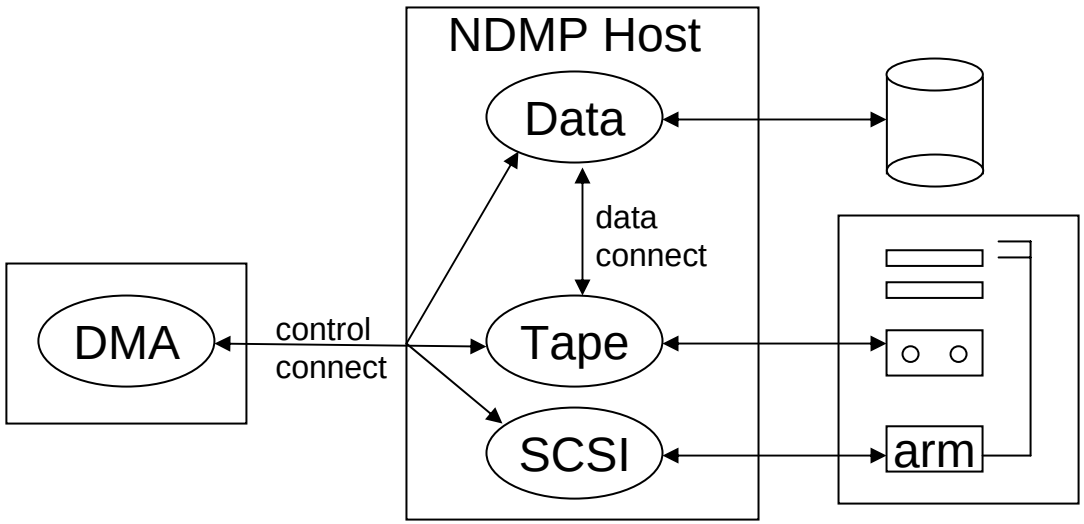
Typically used by the DMA to manipulate a SCSI or fibre channel attached media changer.



Example NDMP Topologies and Configurations

Locally Attached (NDMPv1)

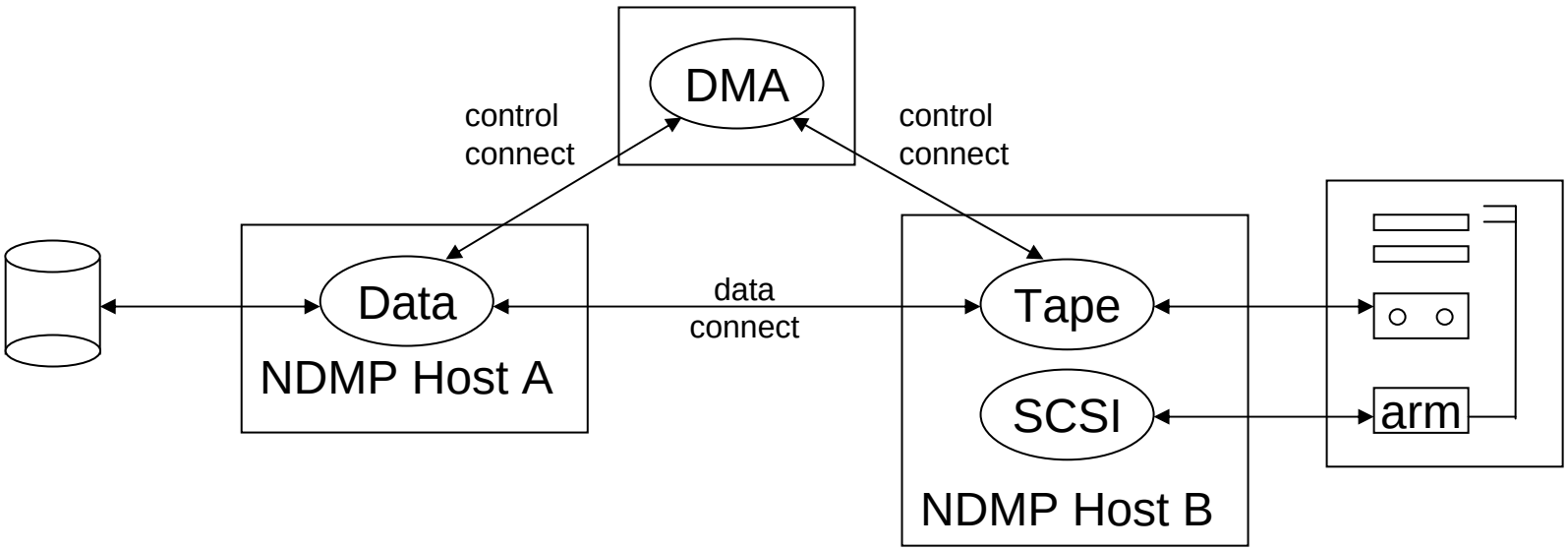
- All NDMP Severs are run from a single NDMP Host
- All of the hardware is attached to the NDMP Host



Example NDMP Topologies and Configuration

3 Way (NDMPv2):

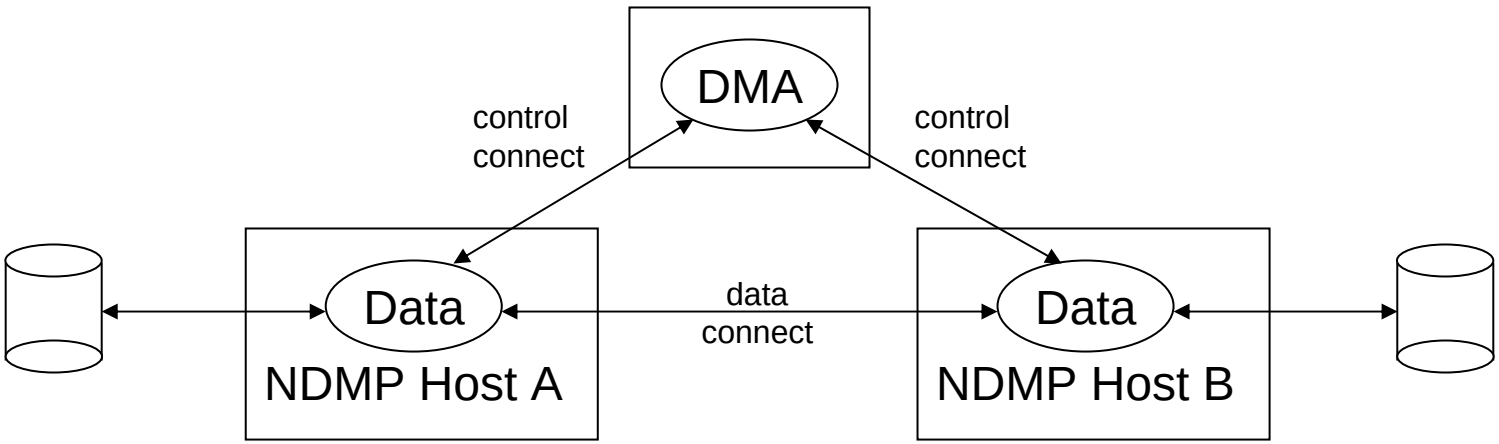
The Data Server on one NDMP Host communicates with the Tape Server on a second NDMP Host



Example NDMP Topologies and Configuration

Data to Data (NDMPv3):

The Data Server on one NDMP Host communicates with the Data Server on a second NDMP Host to perform a data copy.



NDMP Specification: Interfaces (Request/Reply)

An NDMP Server provides a subset of the following interfaces to the DMA:

Connect Interface: Authentication, negotiates the version of protocol

Config Interface: Allows a DMA to discover the configuration of the NDMP Server (includes commands for discovering Extensions)

Data Interface: Initiates backup and recover operations.

- **The DMA provides all the parameters**
- **The DMA does not place constraints on the format of the backup data**

Tape interface: Used to position tapes and to read and write tape metadata such as tape labels

Mover Interface: Controls the reading and writing of backup data from and to a tape device.

SCSI Interface: Allows the DMA to control locally attached media changers.

NDMP Specification: Interfaces (Post)

An NDMP Server implementation MAY send the following messages to the DMA

Notify Interface: Notify the DMA that attention is required.

File History interface: The NDMP Server provides file history information that the DMA uses to track the files contained in each backup and to select and locate files for recovery

DAR: Direct Access Recovery provides faster access to data on tape

Log Interface: Enables the NDMP Server to make entries in the backup and recovery log.

All the messages that the DMA accepts are asynchronous
None of these messages generates a reply message.

NDMP Specification: Interfaces (Extensions)

Develop and standardize new functionality in NDMP without requiring a revision of core NDMP

Expose proprietary functionality in NDMP Server implementations

DMAs can discover and negotiate the use of these extensions

Extensions are managed at two levels:

- **Standard extensions developed or ratified by the NDMP community**
- **Proprietary extensions developed for the individual implementations**

Typically will requires DMAs to add functionality

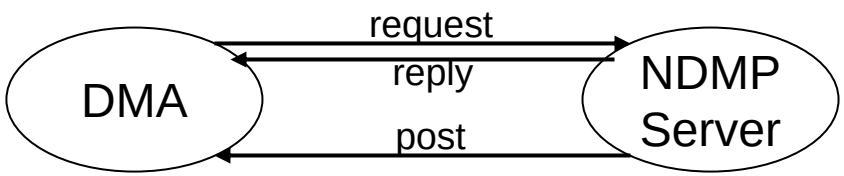
NDMP Specification: Protocol

Request, Reply and Post messages sent over a TCP/IP connection

Request messages are sent from the DMA to the NDMP Server, and have corresponding Reply messages

Post messages are used by the NDMP Server to pass information to the DMA, and hence have no associated reply messages

NDMP uses the RPC/XDR Record Marking (RM) Standard



NDMP Specification: Protocol

An NDMP message consists of a message header optionally followed by a message body:

Each message is XDR encoded and sent within a single RM record.
The message header identifies the message and defines how to de-serialize the arguments and dispatch the message.

sequence: starts at 1 for DMA (Request) and Server (Post)
time_stamp: seconds since 00:00:00 GMT, Jan 1, 1970
message_type: REQUEST or REPLY
message_code: identifies the message
reply_sequence: 0 for requests, request sequence for replies
error_code: 0 for requests, NDMP receive/decode error for replies

Current Backup and Recovery Model for AFS

The current AFS Backup System has some similarities with NDMP:

DMA:

- **backup: some configuration, backup and recovery operations**
- **buserver: maintains the Backup Database which contains configuration information along with records of the tapes created and volumes dumped**

Data Server:

- **vlserver: identifies the current locations of volumes**
- **volserver: file server application that provides acess to volume data**
- **buserver: maintains "dumpdates" info for level (0-9) backups**
- **butc: receives backup requests, backs up data from the volserver**

Tape/SCSI Server:

- **butc: mounts/labels tapes, tracks tape usage, controls tape libraries**

Current Backup and Recovery Model for AFS

But it is not exactly NDMP…

The buserver provides some functionality of the DMA and the Data Server

butc provides some functionality of the Data, Tape, and SCSI Servers
butc hides control of tape devices/robots from the DMA

Volume data is not examined for File History (garbage in, garbage out)

Data stream may not be compatible with NDMP?

Difficult to integrate the NDMP protocol into the existing AFS Backup System

Should be some reusable code

Requirements for an NDMP Server for AFS

Backup Software Vendors provide the DMA

Most NAS filers implement all of the CORE NDMP Servers (Data/Tape/SCSI)
- including Post messages to the DMA (Notify, File History, Log)

Third Party Tape/SCSI Servers available (FastStream NDMP 4600)

Who might want to use NDMP with AFS?

- **Already use NDMP**
- **Use software that supports NDMP**
- **Would like consolidate AFS backups with other data**
- **Want single file/sub-directory restore capability, including DAR**

Requirements for an NDMP Server for AFS

An NDMP Data Server for AFS would provide the fastest integration with existing NDMP solutions

- **Focus on issues specific to AFS file system layout**
- **Define a data stream format compatible with NDMP**
- **Provide File History (including DAR) for partial restores**

Data Server would provide following interfaces:

- **Connect Interface**
- **Config Interface**
- **Data Interface**
- **Post messages to the DMA (Notify, File History, Log)**

Data Server for AFS: Connect Interface (NDMP_CONNECT_)

Authentication and protocol negotiation

- **OPEN (NDMP_CONNECT_OPEN)**
- **CLOSE**
- **CLIENT_AUTH**
- **SERVER_AUTH (optional)**

OPEN:

DMA Request: A protocol version (1, 2, 3 or 4)

Data Server Reply:

NDMP_NO_ERR
NDMP_ILLEGAL_ARGS_ERR
NDMP_NOT_SUPPORTED_ERR
NDMP_ILLEGAL_STATE_ERR

CLOSE:

DMA uses to close the NDMP connection (only request with no reply)

Data Server for AFS: Connect Interface (NDMP_CONNECT_)

CLIENT_AUTH:

NONE: No authentication is required.
TEXT: A user name and a clear text password.
MD5: A user name and a MD5 generated auth_digest derived from a challenge string and a password known to both the DMA and the server.

SERVER_AUTH:

This optional request is used by the DMA to force the NDMP Server to authenticate itself.

An NDMPv4 Extension may be used to add Kerberos 5

At least one of the standard methods (MD5) must be supported

Data Server for AFS: Config Interface (NDMP_CONFIG_)

Allows the DMA to discover the configuration of the NDMP Server.

- **GET_HOST_INFO**
- **GET_SERVER_INFO**
- **GET_CONNECTION_TYPE**
- **GET_AUTH_ATTR**
- **GET_BUTYPE_INFO**
- **GET_FS_INFO**
- **GET_TAPE_INFO***
- **GET_SCSI_INFO***
- **GET_EXT_LIST**

***Not required for an initial NDMP Data Server for AFS**

- **NDMP_NOT_SUPPORTED_ERR**

Data Server for AFS: Config Interface (NDMP_CONFIG_)

GET_HOST_INFO:

Get host information about the NDMP Host.

Server Reply:	NAS (BlueArc)	AFS Examples
hostname<>	ndmp1.domain	afs1.domain
os_type<>	BOS	OpenAFS
os_vers<>	4.1	1.4.4
hostid<>	ae8ef643711a:1	ae8ef643711a:1

Data Server for AFS: Config Interface (NDMP_CONFIG_)

GET_SERVER_INFO:

Get information about the NDMP Server implementation.

Server Reply:	NAS	AFS Examples
vendor_name<>	BlueArc Corp	openafs.org
product_name<>	Silicon Server NDMP	OpenAFS NDMP Data
revision_number<>	4.1	1.4.4
auth_type<>	NDMP4_AUTH_TEXT NDMP4_AUTH_MD5	NDMP4_AUTH_MD5 NDMP4_AUTH_KERB5

Data Server for AFS: Config Interface (NDMP_CONFIG_)

GET_CONNECTION_TYPE:

Returns a list of the data connection types supported by the NDMP Server.

Server Reply: Array of supported connection types

LOCAL: The Data and Tape services are run within the same NDMP Server

TCP: One NDMP Server listens for a TCP/IP connection from another NDMP Server

IPC: Two NDMP Servers on the same host are controlled by separate DMA connections

Data Server for AFS: Config Interface (NDMP_CONFIG_)

GET_AUTH_ATTR:

Obtain the attributes of the authentication methods supported by the server.

MD5: the DMA will use this message to obtain the server challenge string before sending the NDMP_CONNECT_CLIENT_AUTH message.

Server Reply: a 64 byte challenge string.

Data Server for AFS: Config Interface (NDMP_CONFIG_)

GET_BUTYPE_INFO:

Query the backup types supported by the NDMP Server and their capabilities

Server Reply: an array of supported backup types each with

butype_name: Name of the backup application (“dump” or “afs”).

attrs: Backup attributes bit mask, initial implementation for AFS:

- **DIRECT: DAR compatible file history**
- **INCREMENTAL: incremental backup**
- **UTF8: UTF8 format of file names in the file history (ASCII)**
- **FH_DIR: Generate dump style file history (vs. tar)**

Data Server for AFS: Config Interface (NDMP_CONFIG_)

GET_BUTYPE_INFO (continued)

default_env: The default value of the environment variables specific to the backup type.

	NAS (BlueArc)	AFS Examples
FILESYSTEM	NULL	NULL
DIRECT	n	n
TYPE	dump	dump (or afs)
HIST	y	y
EXCLUDE	NULL	(returned of AFS supports omit rules)
LEVEL	0	0
UPDATE	y	y

Data Server for AFS: Config Interface (NDMP_CONFIG_)

GET_FS_INFO:

Query information about the file systems on the NDMP Server host.

Server Reply:
unsupported: The unsupported bit mask (TOTAL_INODES_UNS)
fs_type: "AFS"
fs_logical_device: The mount point of the file system (/root/cell)
fs_physical_device: The physical device name (/serv/part/root/cell)
total_size: maxquota
used_size: diskused
avail_size: maxquota-diskused, vice partition free space
total_inodes: unsupported
used_inodes: filecount

All sizes in bytes, example values obtained from vos examine -format

Data Server for AFS: Config Interface (NDMP_CONFIG_)

GET_FS_INFO (continued):

fs_env: The environment variables defined for the file system

LOCAL: Is volume local to the machine on which the data service is running (might be useful for volume relocation)
TYPE: Kind of file system, “AFS”
AVAILABLE_BACKUP: “dump” or “afs”
AVAILABLE_RECOVERY “dump” or “afs”

fs_status: Should be one of: "online," "offline," or another implementation-specific string (“busy”).

Data Server for AFS: Config Interface (NDMP_CONFIG_)

GET_EXT_LIST:

Request which classes of extensions and versions are available.

Reply: The list of classes of extensions and versions of these that the NDMP Server supports.

SET_EXT_LIST:

Select which extensions, and which version of each extension to use.

Request: The structure lists the classes that the DMA will use.

- **Includes only one instance of each class.**
- **Version is one returned from GET_EXT_LIST**

Data Server for AFS: Data Interface (NDMP_DATA_)

Data Interface: Manages the transfer of backup and recovery stream data between a Tape Server and the file system represented by the local Data Server. There are nine (9) unique request/reply message pairs from the DMA

Connection Management

- **CONNECT: Establish a data connection to peer Data/Tape Server**
- **LISTEN: Accept a data connection from peer Data/Tape Server**
- **ABORT: Terminate unfinished backup/recovery processing**
- **STOP: A normal end of session**

Data Transfer Management

- **START_BACKUP: Start backup operations**
- **START_RECOVER: Start recovery operations**
- **START_RECOVER_FILEHIST (optional)**

Status Reporting

- **GET_STATE: Get backup/recovery progress information**
- **GET_ENV: Get final backup environment variables**

Data Server for AFS: Data Interface (NDMP_DATA_)

START_BACKUP:

DMA Request:
butype_name: “dump” or “afs”
env_vars: from GET_BUTYPE_INFO

	NAS (BlueArc)	AFS Examples
FILESYSTEM	/home/user1	/root/cell
DIRECT	y	y
TYPE	dump	dump (or afs)
HIST	y	y
EXCLUDE	tmp,scratch	(depends on implementation)
LEVEL	2	2
UPDATE	y	y

NOTE: DMA issues GET_ENV to get final backup environment variable values

Data Server for AFS: Data Interface (NDMP_DATA_)

START_RECOVER:

DMA Request:

backup_env<>: from backup (e.g. FILESYSTEM=/root/cell)
nlist<>: source/destination pathname info
butype_name<>: “dump” or “afs”
node: used to indicate a single file for selective file recovery.
fh_info: typically an offset in the backup stream used for DAR

Example: A selective recovery of a directory “mydata” from user.kwebb a new volume named user.kwebb.rest in a directory named “mydata.restored”

FILESYSTEM = "/usr/kwebb"
original_path = “saved"
destination_path = "/usr/kwebb/restore“
name = “mydata.restored”

Open Design Issues (volume management)

Volume Relocation: Cell vs. FileServer implementation

Cell: A single NDMP Data Server for each AFS cell

- **Volume relocation not an issue for DMAs**
- **All backup stream data must pass through a single NDMP Host**
- **DMA providers could optionally redirect backups to a fileserver**

FileServer: An NDMP Data Server for each file server

- **Volume relocation would result in a new full backup**
- **DMA provides could optionally track movement of volumes**
- **Backup stream data sent directly from file server to Tape Server**

New NDMP Environment Variables

AFS_FILESERVER: value of last known server location
AFS_VICE_PARTITION: value of last known partition location

Inconsistencies in vldb and file server states

Open Design Issues (dump/fs format)

AFS incremental dump sends all directory info, file changes, and file tags

NDMP incremental dump sends only directory and file changes

File History generation may require a new approach

DAR recovery may not work well for subdirectory restores, single files are okay (“dump” vs. “afs”)

Open Design Issues (dumpdates management)

AFS Backup System stores dump time information in the Backup Database

A new method for managing dump times may be required

OR use a post backup Environment Variables:

- **AFS_DUMPTIME**
  - **Data Server returns on each successful backup**
  - **DMA sends appropriate time to Data Server for next level backup**

- **Environment Variables are easily tracked by DMAs**

- **Sending the right value may require DMAs to add functionality**

Open Design Issues (snapshots)

Snapshots are well supported by DMAs

Environment Variables easily control this functionality:

AFS_USE_SNAPSHOT: control the use of .backup volumes
AFS_UPDATE_SNAPSHOT: control update of .backup volumes

If AFS_UPDATE_SNAPSHOT=no (vos backupsys run each night), must make sure that proper times are used for incremental backups.

Open Design Issues (Restores)

Database servers/DR Issues

Volume level restore takes a volume offline.

Single files: can we recover to a live volume w/o taking it offline?

DMA redirect of data to a different volume, vice partition, file server, or cell?

Open Design Issues (misc)

Communication protocols: Data Server acts as a translator

Sub-directory/single file backup: Need to be careful about how pathname resolution happens

- **Example: is /root/cell/i386_linux12 a sub directory of root.cell or a separate volume?**

Multi cell support: nothing stands out

Parallel Backups: Cell based Data Server may be a bottleneck, may be a way for the Data Server to contact the correct file server to run the job.

Scale: Number of AFS volumes may perplex some DMAs

Strengths and Weaknesses of an NDMP Solution for AFS

Strengths

- Increased support for AFS from software vendors (DMAs)
- Helps to keep upper level management happy
- Allows AFS to be more easily adopted
- Partial restores, DAR recovery
- Consolidation of backup hardware and software
- Development of OpenAFS
  - more focused backup stream format
  - less concerned about tapes, robots, backup and restore interfaces
- Optional development of Tape/SCSI Server allows flexible configurations

Strengths and Weaknesses of an NDMP Solution for AFS

Weaknesses (AFS specific)

- Poor authentication, Kerberos Extension adoption may be slow
- Does not provide for volume relocation well
- Concepts like “cell” and “vice partition” not well understood by DMAs
- Too many Extensions for AFS may scare software vendors off
- Not clear the DMAs will deal well with 100K individual volumes

Weaknesses (NDMP in general)

- DSPs unwilling to provide clear definition of data stream format
- Cross architecture restore virtually non-existent
- Currently no synthetic backup capability in NDMP
- Disk to Disk backup for NDMP adoption is slow
  - VTLs can be used for disk based backups for NDMP
- No clear verification method for tape backups (data checksums)

Conclusions

An NDMP Data Server for AFS is feasible

Lots of tradeoffs between ease of implementation and acceptance by DMAs

Planned changes in the AFS architecture could impact NDMP design

AFS & Kerberos Best Practices '07

Enjoy the Workshop!